Universidade de Aveiro Departamento de Matemática,
2017

João Pedro Ladeiro
Monteiro da Silva

Recursos digitais de apoio ao ensino de Primitivas e Integrais no Ensino Secundário

Universidade de Aveiro Departamento de Matemática,
2017

João Pedro Ladeiro Monteiro da Silva

Recursos digitais de apoio ao ensino às Primitivas e Integrais no Ensino Secundário

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Matemática para Professores, realizada sob a orientação científica de Maria Paula Lopes dos Reis Carvalho, Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro e co-orientação científica de Luís António Arsénio Descalço, Professor Auxiliar do Departamento de Matemática da Universidade de Aveiro.

o júri / the jury

presidente / president

Professor Doutor João Pedro Antunes Ferreira da Cruz

Professor Auxiliar do Departamento de Matemática da Universidade de Aveiro

vogais / committee

Professor Doutor Paulo Alexandre Silva Pereira

Professor Auxiliar do Departamento de Matemática e Aplicações da Universidade do Minho (Arguente Principal)

Professora Doutora Maria Paula Lopes dos Reis Carvalho

Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro (Orientadora)

agradecimentos

Agradeço reconhecidamente à Doutora Paula Carvalho e ao Doutor Luís Descalço que orientaram a minha tese com muito esforço e paciência. As suas opiniões, críticas e empenho contribuiram inegavelmente para o avanço do meu trabalho.

Aos meus filhos, Bárbara e Gustavo, de quem tanto me orgulho e em cujo sorriso encontrei estímulo para ser preseverante.

À minha mulher, Mafalda, pilar da minha vida e a quem tudo devo, cujo apoio em períodos de dificuldade e hesitação, pemitiu que nunca me desviasse do meu objetivo final.

Palavras-chave

Primitiva, integral, derivada, Siacua, Megua, Sage mathematics, sagemath, CoCalc, exercícios parametrizados

Resumo

Este trabalho tem como objetivo a criação de recursos digitais consistindo num conjunto de exercícios parametrizados para apoiar o estudo autónomo de primitivas no ensino secundário. Em particular, abordam-se os métodos de primitivação no caso de primitivas imediatas, quase-imediatas e por aplicação do método de primitivação por partes. A utilização destes conteúdos é feita via internet em `http://siacua.web.ua.pt/`. A construção e resolução dos exercícios foi efetuada de acordo com as metas curriculares do programa de matemática de 12.^{o} ano que entra em vigor no ano letivo 2017/2018. Os exercícios constituem uma ferramenta de trabalho tanto para os alunos no seu estudo autónomo e auto-avaliação como para o professor como forma de diagnosticar as lacunas predominantes nas aquisição de conhecimentos dos seus alunos.

Keywords Integral, derivative, Siacua, Megua, Sage mathematics, sagemath, CoCalc, Parametrized Exercises

Abstract The main goal of this work is to create digital resources based in a set of parameterized exercises to support self study of Antiderivatives in Secondary School. This resources are available at `http://siacua.web.ua.pt/`. In this work we had in consideration the curricular guidelines for the 12^{th} year curriculum that began in the curricular year of 2017/2018. The main objective is to provide some elements in order to help students to be autonomous in their study, as well as teachers in the diagnose of students knowledge.

Conteúdo

Lista de Figuras

Lista de Tabelas

Introdução

Desde os finais do século passado que temos vindo a assistir a uma crescente evolução a nível das tecnologias de informação e comunicação, iniciada com o aparecimento dos primeiros telemóveis, no fim dos anos noventa, em substituição dos *pagers*. Por essa altura, a *internet* dava os seus primeiros passos e era quase impensável que, em pouco mais de vinte anos, nos víssemos rodeados de uma tão grande diversidade de dispositivos eletrónicos como *laptops*, *tablets*, *e-books*, e *smartphones*. Com efeito, a grande inovação que parecia ser um telemóvel com teclas, ligado a uma qualquer rede móvel, rapidamente se tornou obsoleta e a constante evolução tecnológica não pára de nos surpreender. A massificação dos recursos informáticos, a facilidade do seu manuseamento e a velocidade de ligação que os caracterizam fazem com que estejamos hoje quase sempre disponíveis, na maior parte dos casos com ligação remota e à distância de um clique. O *mundo digital* instalou-se entre nós e a *internet*, veículo de transmissão de informação em tempo real, tornou-se parte integrante das nossas vidas e impôs-se como fenómeno transversal a diferentes gerações.

Esta nova realidade transpôs-se para as escolas e está presente no perfil do aluno que as frequenta. Não raras vezes vemos estudantes em viagens, de rede viária ou ferroviária, que acedem a redes móveis de *internet* para consultar um qualquer conteúdo do seu interesse. Por outro lado, os alunos do Ensino Secundário recorrem frequentemente a plataformas informáticas e aplicações móveis, criadas por editoras de livros ou pelas próprias escolas, com o intuito de aceder a conteúdos lecionados nas diferentes disciplinas, o que torna necessário que as metodologias e técnicas de ensino/aprendizagem acompanhem a permanente atualização tecnológica a que assistimos. No caso concreto do Ensino de Matemática, a utilização das novas tecnologias constituiu-se uma mudança de paradigma na forma de transmitir conteúdos. Depois do advento das calculadoras gráficas, começaram a surgir programas de computador de apoio ao ensino da disciplina tais como *Cabri Geometre*, *SketchPad* e *GeoGebra*, entre outros. Paralelamente, a criação de competições matemáticas como o concurso *EQUAmat*, da Universidade de Aveiro, veio fomentar o uso das novas tecnologias na consolidação de aprendizagens. Este projeto, que começou de forma tímida, rapidamente se tornou um sucesso a nível nacional e o número de alunos participantes é maior em cada edição realizada.

O *Programa e Metas Curriculares de Matemática A para o Ensino Secundário* surgiu no âmbito da revisão do Currículo Nacional em 2011, cujo sentido é o de elevar os padrões de desempenho escolar dos alunos portugueses, dando continuidade ao *Programa de Matemática para o Ensino Básico.* Com o objetivo de desenvolver no aluno o gosto por esta disciplina milenar, nas suas diversas vertentes, o documento estabelece um conjunto de conhecimentos e de capacidades essenciais que os alunos devem adquirir e desenvolver no decurso do Ensino Secundário, na disciplina de Matemática A, sendo denominadas por *Metas Curriculares* [10]. Segundo os seus autores, o Programa foi concebido por forma a fornecer aos alunos instrumentos que garantam um prosseguimento de estudos com sucesso, tendo em consideração que é o programa de Matemática A do Ensino Secundário que dá acesso aos cursos do Ensino Superior de áreas que requerem uma sólida formação matemática. Com este propósito, foram introduzidos alguns conteúdos fundamentais que se encontravam ausentes no anterior Programa e cujo estudo é recomendado, pelas melhores práticas internacionais, nos ramos do Ensino Secundário com estas características, como é o caso do Cálculo Integral [10]. Um dos bons bons exemplos desta prática é o currículo *Further mathematics High Level* do *International Baccalaureate* [16].

É neste contexto que surge a unidade *Primitivas e Cálculo Integral* como um dos sete domínios de conteúdos que integram o atual programa curricular do 12.º ano de Matemática A. O domínio está denominado como *PCI* e nele são estabelecidas vinte e sete *Metas Curriculares* [10]. A unidade inicia-se com a introdução da definição de primitiva de uma função e o estudo de algumas propriedades. É depois abordada a noção de integral de uma função contínua e não negativa num intervalo limitado, de forma intuitiva e visual, recorrendo à noção de área e, a partir de propriedades elementares admitidas para esta noção, demonstra-se o *Teorema Fundamental do Cálculo* e a *Fórmula de Barrow.* Posteriormente, estende-se a definição às funções contínuas que alternam de sinal um número finito de vezes, bem como os referidos resultados fundamentais. Finalmente, refere-se apenas a possibilidade de extensão a todas as funções contínuas [1, 3, 7, 10]. Em traços gerais, este domínio de conteúdos visa o estudo das principais propriedades dos integrais definidos e a análise de algumas técnicas de primitivação e integração [10]. Desta estruturação, depreende-se a relevância atribuída ao facto de os alunos terminarem o Ensino Secundário com algumas noções, ainda que não inteiramente formalizadas, de Cálculo Integral, já que, em certo sentido, se trata de um complemento essencial do Cálculo Diferencial [10]. Na senda deste objetivo, foram várias as editoras que elaboraram e puseram à disposição dos professores um conjunto de aplicações interativas, em suporte digital, a ser utilizado em contexto de sala de aula ou fora dela. A utilização de novas tecnologias no Ensino de Matemática é reconhecidamente um catalisador da aprendizagem, tanto pela forma como pela atratividade natural dos jovens portugueses pelas novas tecnologias [1, 3, 7].

Nesta conjuntura, o nosso trabalho pretende constituir-se, também ele, uma ferramenta digital de trabalho para professores e alunos no domínio *Primitivas e Cálculo Integral*, integrando-se num projeto mais vasto do Departamento de Matemática da Universidade de Aveiro [6, 13] que visa a criação de uma base de dados de exercícios parametrizados. Desta forma, o nosso objetivo primordial é o de contribuir para a consolidação de aprendizagens com recurso às novas tecnologias de informação e comunicação, ao mesmo tempo que se aborda um conteúdo comum aos Ensinos Secundário e Superior.

Assim alicerçada, esta dissertação apresenta-se organizada em três capítulos.

No Capítulo 1, intitulado *Primitivas*, é apresentada a definição de primitiva, propriedades e algumas técnicas de primitivação.

No Capítulo 2, intitulado *Cálculo Integral*, faz-se uma breve resenha histórica sobre esta temática, aborda-se o *Teorema Fundamental do Cálculo*, apresenta-se a definição de integral definido e trata-se a área de regiões planas.

No Capítulo 3, intitulado *Construção de um exercício*, é feita uma descrição detalhada e sequencial dos procedimentos inerentes à construção de um exercício recorrendo aos *softwares* matemáticos *CoCalc* e *MEGUA* e posterior visualização no sistema *SIACUA*, explicando-se inicialmente em que consistem e como funcionam estes sistemas para depois partir do caso geral que é a construção de um qualquer exercício rumo à construção de um exercício específico.

Na Conclusão deste trabalho redesenha-se uma configuração do ensino do domínio de conteúdos *Primitivas e Cálculo Integral* no Ensino Secundário, são identificadas algumas limitações à sua execução e é feito um balanço geral do nosso trabalho.

No Apêndice A apresentamos os exercícios criados ao longo da duração deste trabalho com o código completo da sua construção.

Capítulo 1

Primitivas

1.1 Definição e Propriedades

Seja I um intervalo de $\mathbb{R}$ ($I \subseteq \mathbb{R}$) que contenha mais que um ponto.

Definição 1.1. [12] Chama-se *primitiva* de $f : I \to \mathbb{R}$ a qualquer função $F : I \to \mathbb{R}$ tal que $F^{'}(x) = f(x)$, para qualquer $x \in I$. Diz-se que f é *primitivável* em I quando possui pelo menos uma primitiva em I.

É imediato a partir da Definição 1.1 que qualquer primitiva de f em I é contínua em I. De facto, qualquer função F naquelas condições é diferenciável em I e, portanto, é contínua em I.

Observação: Esta definição pode ser generalizada para funções definidas em um conjunto $D \subseteq \mathbb{R}$ que não são intervalos. Diz-se que f é primitivável em $I \subseteq D$ se a restrição de f a I é primitivável em I.

Se F é primitiva de f em $I \subseteq \mathbb{R}$ então $F + C$ é também primitiva de f em I para qualquer número real C e para qualquer $x \in I$. Como C é um número real qualquer, se f tem uma primitiva em I, então tem uma infinidade de primitivas em I cujo conjunto constitui a família de primitivas de f. Além disso, a derivada da diferença de duas primitivas de f, F_1 e F_2, em I é nula. $(F_1 - F_2)^{'} = (F_1)^{'} - (F_2)^{'} = f - f = 0$ de onde se pode concluir que F_1 e F_2 diferem de uma constante.

Se $f : I \to \mathbb{R}$ é primitivável em I, denota-se por $\displaystyle\int f(x)\, dx$, ou simplesmente por $\displaystyle\int f$, qualquer uma das primitivas de f em I.

As operações com primitivas têm algumas analogias com as operações com derivadas. O teorema seguinte resulta da linearidade da derivada.

Teorema 1.1. *Se f e g forem funções primitiváveis num intervalo I e k uma constante real então,*

(a) kf é primitivável em I e $\displaystyle\int kf(x)\,dx = k\int f(x)\,dx$.

(b) $f+g$ é primitivável em I e $\displaystyle\int (f+g)(x)\,dx = \int f(x)\,dx + \int g(x)\,dx$.

Este teorema pode generalizar-se para qualquer número finito de funções primitiváveis, isto é, sendo $f_1, f_2, \ldots, f_n$, n funções primitiváveis num intervalo I e sendo $k_1, k_2, \ldots, k_n$, n constantes tem-se que qualquer combinação linear da forma $k_1f_1 + k_2f_2 + \ldots + k_nf_n$ é primitivável em I e além disso,

$$\int (k_1f_1 + k_2f_2 + \ldots + k_nf_n) = k_1\int f_1 + k_2\int f_2 + \ldots + k_n\int f_n$$

No entanto, nem todas as funções são primitiváveis (em todo o seu domínio). Em particular, as que não satisfazem as condições do Teorema do Valor Intermédio num intervalo I do seu domínio: Dados dois pontos $a, b \in I$ tais que $f(a) \neq f(b)$, tomando um qualquer valor k entre $f(a)$ e $f(b)$ existe pelo menos um c entre a e b tal que $f(c) = k$ [4].

Pode provar-se que a derivada de qualquer função diferenciável num intervalo é uma função que verifica o Teorema do Valor Intermédio [4].

Seja $f : \mathbb{R} \to \mathbb{R}$ definida por

$$f(x) = \begin{cases} 1 & se \quad x \geqslant 0 \\ 0 & se \quad x < 0 \end{cases}$$

Esta função não tem primitiva em intervalos do tipo $]a, b]$ com $a < 0 \leqslant b$. Mas, em qualquer outro intervalo $]a, b]$, a primitiva existe e é da forma:

$$F(x) = x + C \text{ se } a \geqslant 0 \quad \text{ou} \quad F(x) = C \text{ se } b < 0,\ \ C \in \mathbb{R}.$$

1.2 Técnicas de primitivação

A relação natural entre a primitivação e a derivação tem como consequência que o cálculo das primitivas de certas funções é imediato, usando o conhecimento das regras de derivação.

Nesta secção vamos expor algumas técnicas de primitivação de funções a que se chamam funções elementares, ou seja, as funções polinomiais, exponenciais, trigonométricas e as suas inversas assim como todas as que se podem obter a partir destas por aplicação de um número finito de operações de adição, multiplicação, divisão, radiciação e composição.

Por exemplo, são funções elementares as funções definidas por expressões do tipo

$$1+x^2,\ x\sin\left(x^2-1\right),\ \ln\left(x\right)+3^x,\ \sqrt{\sqrt{\frac{\tan\left(x^2+1\right)}{\ln\left(x^2\right)}}+x}.$$

No ensino secundário apenas é tratada a chamada primitivação imediata e quase imediata, isto é, funções definidas por expressões do tipo: 1, x^α, $(\alpha\in\mathbb{R}\backslash\{0,-1\})$, $\frac{1}{x}$, $\sin x$ e $\cos x$ [10]. No entanto, neste texto tratamos também outras técnicas sem pretender um aprofundamento demasiado exaustivo.

1.2.1 Primitivas imediatas e quase imediatas

A primitivação imediata consiste, de um modo informal, na aplicação das regras de derivação em sentido inverso. Na tabela 1.2.1 estão algumas dessas regras.

Raramente as funções a primitivar aparecem exatamente nesta forma. Muitas vezes as funções a primitivar resultam de composição de funções elementares de tal modo que é possível aplicar as regras de derivação de modo quase imediato. Por exemplo, sendo $f\left(x\right)=g\left(u\left(x\right)\right)$ a função obtida por composição das funções f e u, sabemos que $f'\left(x\right)=g'\left(u\left(x\right)\right)u'\left(x\right)$ por aplicação da regra da derivada da função composta (tabela 1.2.1)

Assim,

$$\int g'\left(u\left(x\right)\right)u'\left(x\right)dx=f\left(x\right)+C,\ C\in\mathbb{R}.$$

Tabela 1.2.1: Primitivas imediatas

Derivadas	Primitivas
$\left(x^{\alpha+1}\right)' = (\alpha+1)\, x^{\alpha}, \alpha \neq 1$	$\displaystyle\int x^{\alpha} dx = \frac{1}{\alpha+1} x^{\alpha+1} + C, \alpha \neq 1$
$(\ln x)' = \dfrac{1}{x}$	$\displaystyle\int \frac{1}{x} dx = \ln\lvert x\rvert + C$
$(e^{x})' = e^{x}$	$\displaystyle\int e^{x} dx = e^{x} + C$
$(\sin x)' = \cos x$	$\displaystyle\int \cos x dx = \sin x + C$
$(\cos x)' = -\sin x$	$\displaystyle\int \sin x dx = -\cos x + C$

Exemplo 1.1. Calcular $\displaystyle\int x\, e^{3\,x^2} dx$.

Fazendo $g(u) = e^{u}$ e $u(x) = 3\,x^2$, aplicando a fórmula anterior e multiplicando por 6 e pelo seu inverso (para assim obtermos $u'(x)$) temos:

$$\begin{aligned}
\int x e^{3\,x^2} dx &= \frac{1}{6} \int g'(u(x))\, u'(x)\, dx \\
&= \frac{1}{6} \int 6\, x\, e^{3\,x^2} dx \\
&= \frac{1}{6} \left(e^{3\,x^2} + C \right), \ C \in \mathbb{R}.
\end{aligned}$$

Chamamos primitivas *imediatas* às que resultam da aplicação da tabela 1.2.1 e chamamos primitivas *quase imediatas* às que resultam da derivação da função composta às quais aplicamos as regras da mesma tabela.

Convém sublinhar que primitivação imediata não é sinónimo de que a função seja simples ou fácil de primitivar. Em certos casos é necessária uma considerável manipulação algébrica e uma certa dose de cuidado.

Exemplo 1.2. Calcular $\int \left(x^2-1\right)^3 dx$.

Para calcular $\int \left(x^2-1\right)^3 dx$ pode usar-se o desenvolvimento binomial (Binómio de Newton), $\left(x^2-1\right)^3 = x^6 - 3x^4 + 3x^2 - 1$, através do qual se obtém de modo agora imediato,

$$\begin{aligned}\int \left(x^2-1\right)^3 dx &= \int \left(x^6 - 3x^4 + 3x^2 - 1\right) dx \\ &= \frac{x^7}{7} - \frac{3x^5}{5} + x^3 - x + C,\ C \in \mathbb{R}.\end{aligned}$$

Exemplo 1.3. Calcular $\int \frac{x^3}{2+x^4}\, dx$.

Para calcular $\int \frac{x^3}{2+x^4}\, dx$ basta observar que multiplicando o numerador da fração por 4, obtém-se uma função do tipo $\frac{u'(x)}{u(x)}$, onde $u(x) = 2 + x^4$.

Assim,

$$\begin{aligned}\int \frac{x^3}{2+x^4}\, dx &= \frac{1}{4}\int \frac{4x^3}{2+x^4}\, dx \\ &= \frac{1}{4}\ln\left(2+x^4\right) + C,\ C \in \mathbb{R}.\end{aligned}$$

Exemplo 1.4. Calcular $\int \frac{1}{x}\, dx$.

Este exemplo corresponde ao caso de $\int x^\alpha\, dx$ quando $\alpha = -1$. Tem-se

$$\ln|x| = \begin{cases} \ln(x) & se \quad x > 0 \\ \ln(-x) & se \quad x < 0 \end{cases}.$$

Assim,

$$(\ln|x|)' = \begin{cases} \dfrac{x'}{x} & se \quad x > 0 \\ \\ \dfrac{-x'}{-x} & se \quad x < 0 \end{cases} = \begin{cases} \dfrac{1}{x} & se \quad x > 0 \\ \\ \dfrac{-1}{-x} & se \quad x < 0 \end{cases} = \frac{1}{x} \qquad \text{se } x \neq 0$$

Portanto, $\displaystyle\int \frac{1}{x}\, dx = \ln|x| + C,\ C \in \mathbb{R},\ x \in \mathbb{R}\backslash\{0\}$

Tabela 1.2.2: Primitivas imediatas de funções compostas

$\displaystyle\int ku = k\int u$	$\displaystyle\int a^u u' = \frac{a^u}{\ln a}$
$\displaystyle\int u^\alpha u' = \frac{u^{\alpha+1}}{\alpha+1}$	$\displaystyle\int u' \sin u = -\cos u$
$\displaystyle\int \frac{u'}{u} = \ln|u|$	$\displaystyle\int u' \cos u = \sin u$
$\displaystyle\int \frac{u'}{2\sqrt{u}} = \sqrt{u}$	$\displaystyle\int \frac{u'}{\cos^2 u} = \tan u$
$\displaystyle\int e^u u' = e^u$	

A tabela das derivadas pode ser obtida por simples troca dos termos das igualdades.

1.2.2 Primitivação de funções racionais

Designa-se por *fração racional* toda a fração cujo numerador e denominador são polinómios, isto é, uma fração do tipo $f(x) = \dfrac{P_1(x)}{P_2(x)}$.

Uma *função racional* é uma função $f : D \to \mathbb{R}$ definida por

$$f(x) = \frac{P_1(x)}{P_2(x)},$$

onde $D = \{x \in \mathbb{R} : P_2(x) \neq 0\}$ [11].

Se o grau de $P_1(x)$ é estritamente inferior ao grau de $P_2(x)$ dizemos que $\dfrac{P_1(x)}{P_2(x)}$ é uma *fração racional própria*. Se o grau de $P_1(x)$ é maior ou igual que o grau de $P_2(x)$ então, fazendo a divisão de $P_1(x)$ por $P_2(x)$ temos

$$P_1(x) = Q(x)P_2(x) + R(x) \Leftrightarrow \frac{P_1(x)}{P_2(x)} = Q(x) + \frac{R(x)}{P_2(x)},$$

onde $Q(x)$ e $R(x)$ são polinómios e o grau de $R(x)$ é menor que o grau de $P_2(x)$, ou seja, $\dfrac{R(x)}{P_2(x)}$ é uma fração racional própria.

Desta forma reduzimos o estudo do cálculo de primitivas de funções racionais ao caso de frações próprias.

A ideia pode ser exposta em duas fases[12]:

A primeira consiste em saber primitivar as frações simples. Chama-se fração simples a qualquer fração racional da forma $\dfrac{A}{(x-\alpha)^r}$ com A uma constante real.

A segunda consiste em decompor a fração racional na soma de um polinómio com frações simples (sendo os polinómios fáceis de primitivar, o problema torna-se menos complexo).

Primitivação de frações simples

Para primitivar $\dfrac{A}{(x-\alpha)^r}$ vamos considerar dois casos:

1.º Caso: Se $r = 1$,

$$\int \frac{A}{(x-\alpha)^r}\,dx = \int \frac{A}{x-\alpha}\,dx = A\ln|x-\alpha| + C,\ \ C \in \mathbb{R}$$

em qualquer intervalo I de $\mathbb{R}$ tal que $\alpha \notin I$.

2.º Caso: Se $r \neq 1 \wedge r > 0$, trata-se de calcular a primitiva de uma potência de expoente negativo.

$$\begin{aligned}\int \frac{A}{(x-\alpha)^r}\,dx &= A\int (x-\alpha)^{-r}\,dx \\ &= A\,\frac{(x-\alpha)^{-r+1}}{-r+1} \\ &= -\frac{A}{r-1} \times \frac{1}{(x-\alpha)^{r-1}} + C,\ \ C \in \mathbb{R}.\end{aligned}$$

em qualquer intervalo I de $\mathbb{R}$ tal que $\alpha \notin I$.

Decomposição de uma fração racional em frações simples

O processo baseia-se no seguinte teorema cuja demonstração pode ser vista em [12].

Teorema 1.2. *Qualquer fração racional pode ser decomposta na soma de um polinómio com frações simples.*

Seja $R(x) = \dfrac{P_1(x)}{Q_1(x)}$ uma fração racional. Para primitivar $R(x)$ vamos considerar três casos:

1.º Caso: O polinómio do denominador admite raízes reais simples $a_1, a_2, \ldots, a_n$.

Efetua-se a decomposição da fração na soma de n frações simples cujos numeradores são constantes a determinar e cujos denominadores são $x-a_1, x-a_2, \ldots, x-a_n$ respetivamente.

$$R(x) = \frac{A_1}{x-a_1} + \frac{B_1}{x-a_2} + \cdots + \frac{B_n}{x-a_n}$$

Exemplo 1.5. Calcular $\displaystyle\int \frac{x^2}{x^2-1}dx$.

Comecemos por notar que $\dfrac{x^2}{x^2-1} = 1 + \dfrac{1}{x^2-1}$ e $\dfrac{1}{x^2-1}$ é uma fração própria que vamos decompor na soma de duas frações simples.

Tendo em conta que as raízes do denominador são 1 e -1 tem-se,

$$\frac{1}{x^2-1} = \frac{1}{(x-1)(x+1)} = \frac{A}{x-1} + \frac{B}{x+1}$$

ou seja,

$$\frac{1}{(x-1)(x+1)} = \frac{A(x+1)+B(x-1)}{(x-1)(x+1)}$$

que, pela igualdade entre frações vem,

$$1 = A(x+1) + B(x-1) \Leftrightarrow 1 = Ax + A + Bx - B \Leftrightarrow 1 = (A+B)x + (A-B)$$

Igualando os coeficientes dos polinómios,

$$\begin{cases} A+B &= 0 \\ A-B &= 1 \end{cases} \Leftrightarrow \begin{cases} A &= -B \\ -2B &= 1 \end{cases} \Leftrightarrow \begin{cases} A &= \frac{1}{2} \\ B &= -\frac{1}{2} \end{cases}.$$

Então,

$$\frac{1}{x^2-1} = \frac{1}{(x-1)(x+1)} = \frac{\frac{1}{2}}{x-1} + \frac{-\frac{1}{2}}{x+1}.$$

Desta forma,

$$\begin{aligned}
\int \frac{x^2}{x^2-1}\,dx &= \int \left(1+\frac{1}{x^2-1}\right)dx \\
&= \int 1\,dx + \int \frac{\frac{1}{2}}{x-1}\,dx - \int \frac{\frac{1}{2}}{x+1}\,dx \\
&= x + \frac{1}{2}\int \frac{1}{(x-1)}dx - \frac{1}{2}\int \frac{1}{(x+1)}dx \\
&= x + \frac{1}{2}\ln|x-1| - \frac{1}{2}\ln|x+1| + C, \quad C \in \mathbb{R}.
\end{aligned}$$

2.º caso: O polinómio do denominador admite raízes reais múltiplas. Neste caso, a cada raiz real a de multiplicidade m corresponde a soma das frações simples:

$$\frac{A_1}{x-a} + \frac{A_2}{(x-a)^2} + \cdots + \frac{A_m}{(x-a)^m},$$

com A_1, A_2, ..., A_m, constantes a determinar.

Exemplo 1.6. Determinar $\displaystyle\int \frac{x+1}{x^3(x+2)}\,dx$.

As raízes do polinómio do denominador são -2, raiz real simples, e 0, raiz real de multiplicidade 3. Assim,

$$\frac{x+1}{x^3(x+2)} = \frac{A}{x^3} + \frac{B}{x^2} + \frac{C}{x} + \frac{D}{x+2}$$

ou seja,

$$\frac{x+1}{x^3(x+2)} = \frac{A(x+2)+Bx(x+2)+Cx^2(x+2)+Dx^3}{x^3(x+2)},$$

pelo que,

$$x+1 = A(x+2)+Bx(x+2)+Cx^2(x+2)+Dx^3.$$

Reescrevendo o polinómio do segundo membro da igualdade na forma canónica,

$$\begin{aligned}
x+1 &= A(x+2)+Bx(x+2)+Cx^2(x+2)+Dx^3 \\
&= Ax+2A+Bx^2+2Bx+Cx^3+2Cx^2+Dx^3 \\
&= (C+D)x^3+(B+2C)x^2+(A+2B)x+2A,
\end{aligned}$$

e igualando os coeficientes dos polinómios,

$$\begin{cases} C+D &= 0 \\ B+2C &= 0 \\ A+2B &= 1 \\ 2A &= 1 \end{cases} \Leftrightarrow \begin{cases} D &= \frac{1}{8} \\ C &= -\frac{1}{8} \\ B &= \frac{1}{4} \\ A &= \frac{1}{2} \end{cases}$$

Então,

$$\frac{x+1}{x^3(x+2)} = \frac{\frac{1}{2}}{x^3} + \frac{\frac{1}{4}}{x^2} + \frac{-\frac{1}{8}}{x} + \frac{\frac{1}{8}}{x+2}$$

Logo,

$$\begin{aligned} \int \frac{x+1}{x^3(x+2)}\, dx &= \int \frac{\frac{1}{2}}{x^3} + \frac{\frac{1}{4}}{x^2} + \frac{-\frac{1}{8}}{x} + \frac{\frac{1}{8}}{x+2}\, dx \\ &= \frac{1}{2}\int x^{-3}\, dx + \frac{1}{4}\int x^{-2}\, dx - \frac{1}{8}\int \frac{1}{x}\, dx + \frac{1}{8}\int \frac{1}{x+2}\, dx \\ &= -\frac{1}{4x^2} - \frac{1}{4x} - \frac{1}{8}\ln|x| + \frac{1}{8}\ln|x+2| + C, \quad C \in \mathbb{R}. \end{aligned}$$

3.º caso: O denominador é um polinómio que tem raízes complexas.

Vamos subdividir este **3.º caso** em três situações:

(i) A situação mais simples é $R(x) = \dfrac{1}{1+x^2}$ cujas raízes são $-i$ e i. Neste caso, a primitiva é imediata (ver Tabela 1.2.1):

$$\int \frac{1}{1+x^2} dx = \arctan x + C, \quad C \in \mathbb{R}.$$

(ii) Outra situação simples é quando temos no denominador um polinómio de grau 2 irredutível (do tipo ax^2+bx+c, com $\Delta = b^2-4ac < 0$).

Neste caso, usando a técnica de completamento do quadrado, o denominador pode escrever-se como $\alpha\left(1+(\beta x+\gamma)^2\right)$, para constantes apropriadas α, β e γ.

Exemplo 1.7. Calcular $\displaystyle\int \frac{1}{x^2-2x+2}\, dx$.

Como $x^2-2x+2 = x^2-2x+1+2-1 = 1+(x-1)^2$, tem-se

$$\frac{1}{x^2-2x+2} = \frac{1}{1+(x-1)^2}$$

e, portanto, como se viu em **(i)**

$$\int \frac{1}{x^2-2x+2}dx = \int \frac{1}{1+(x-1)^2}dx = \arctan(x-1)+C, \quad C \in \mathbb{R}.$$

O exemplo seguinte ilustra como se procede no caso de o polinómio do denominador ter raízes reais e raízes não reais.

Exemplo 1.8. Pretendemos calcular $\displaystyle\int \frac{x^2+x+2}{(x-1)(x^2+1)}\,dx$

O polinómio do denominador já está fatorizado (em $\mathbb{R}$). Podemos escrever a função integranda na forma:

$$\frac{x^2+x+2}{(x-1)(x^2+1)} = \frac{A}{x-1} + \frac{Bx+D}{x^2+1}$$

donde

$$\begin{aligned} x^2+x+2 &= A(x^2+1)+(Bx+D)(x-1) \\ &= Ax^2+A+Bx^2-Bx+Dx-D \\ &= (A+B)x^2+(D-B)x+A-D. \end{aligned}$$

Igualando os coeficientes dos polinómios,

$$\begin{cases} A+B &= 1 \\ D-B &= 1 \\ A-D &= 2 \end{cases} \Leftrightarrow \begin{cases} A &= 1-B \\ D &= 1+B \\ -2B &= 2 \end{cases} \Leftrightarrow \begin{cases} A &= 2 \\ D &= 0 \\ B &= -1 \end{cases}.$$

Portanto,

$$\begin{aligned} \int \frac{x^2+x+2}{(x-1)(x^2+1)}\,dx &= 2\int \frac{1}{x-1}dx - \int \frac{x}{x^2+1}\,dx \\ &= 2\ln|x-1| - \frac{1}{2}\ln(x^2+1)+C, \quad C \in \mathbb{R}. \end{aligned}$$

(iii) O caso em que o denominador é uma potência de um polinómio do segundo grau irredutível, isto é, é da forma $(1+x^2)^n$, para $n \in \mathbb{N}$, é tratado mais adiante neste texto uma vez que recorre à técnica de primitivação por partes.

1.2.3 Primitivação por partes

O método de primitivação por partes não é parte integrante do atual programa de Matemática do Ensino Secundário. Porém, a sua base é a fórmula da derivada do produto, o que torna este método acessível e interessante.

Teorema 1.3. *[12] Se u e v são funções diferenciáveis em I, o produto $u'\,v$ é primitivável em I se e só se o produto $u\,v'$ o for, e tem-se:*

$$\int u'(x)\,v(x)\,dx = u(x)\,v(x) - \int (u(x)\,v'(x))\;dx.$$

Demonstração:

Pela regra da derivada do produto vem $(u(x)\,v(x))' = u'(x)\,v(x) + u(x)\,v'(x)$. Esta igualdade, escrita em ordem a $u'(x)\,v(x)$, fica:

$$u'(x)\,v(x) = (u(x)\,v(x))' - u(x)\,v'(x).$$

Primitivando ambos os membros desta igualdade,

$$\int u'(x)\,v(x)dx = u(x)\,v(x) - \int u(x)\,v'(x)\,dx + C, \quad C \in \mathbb{R}.$$

o que prova o resultado.

Esta expressão é útil na medida em que, em certos casos, escolhendo adequadamente as funções $u(x)$ e $v(x)$ a primitivar, $\displaystyle\int u(x)\,v'(x)dx$ pode calcular-se facilmente enquanto $\displaystyle\int u'(x)\,v(x)\,dx$ não é fácil de calcular.

Exemplo 1.9. Calcular $\displaystyle\int \ln x\,dx$.

Fazendo $u'(x) = 1$ e $v(x) = \ln x$, tem-se $u(x) = x$ e $v'(x) = \dfrac{1}{x}$ donde,

$$\int \ln x\,dx = \int 1 \times \ln x\;dx = x\,\ln x - \int x \times \frac{1}{x}\,dx + C, \quad C \in \mathbb{R}.$$

Portanto, $\int \ln x = x \ln x - x + C$, com $x \in]0, +\infty[$ e $C \in \mathbb{R}$.

Exemplo 1.10. Calcular $\int x^2 \sin x \, dx$.
Fazendo $u'(x) = \sin x$ e $v(x) = x^2$ tem-se:

$$\int x^2 \sin x \, dx = -x^2 \cos x + \int 2x \cos x \, dx.$$

Fazendo uma segunda primitivação por partes temos:

$$\begin{aligned} \int x^2 \sin x \, dx &= -x^2 \cos x + 2x \sin x - 2 \int \sin x \, dx \\ &= -x^2 \cos x + 2x \sin x + 2 \cos x + C, \quad C \in \mathbb{R}. \end{aligned}$$

Usando um método semelhante podemos calcular $\int P(x) cosx \, dx$, $\int P(x) sinx \, dx$ e $\int P(x) \, e^x \, dx$, sendo $P(x)$ uma função polinomial.

Por exemplo, se $P(x)$ for um polinómio de grau p e m um número real diferente de zero então,

$$\int P(x) \, e^{mx} \, dx = e^{mx} \left[\frac{1}{m} P(x) - \frac{1}{m^2} P'(x) + \ldots + (-1)^p \frac{1}{m^{p+1}} P^{(p)}(x) \right]$$

1.2.4 Primitivação por substituição

O método de primitivação por substituição consiste na determinação de uma função invertível $x = g(t)$

$$t \to x = g(t)$$

de modo que na nova variável t a primitiva seja mais fácil de calcular.

Geralmente esta técnica é aplicável quando fazendo uma mudança de variável em $\int f(x) \, dx$, ou seja, definindo uma função diferenciável e invertível $x = g(t)$ podemos escrever

$$\int f(x) \, dx = \int f \left[g(t) \right] g'(t) \, dt$$

e esta primitiva, na variável t é mais fácil de calcular que a inicial. À primitiva obtida na variável t deve aplicar-se a inversa, fazendo $t = g^{-1}(x)$ de modo a regressar à variável inicial.

Exemplo 1.11. Calcular $\int \sqrt{a^2-x^2}$ no seu domínio, isto é, em $I=[-a,a]$, com $a>0$.

Seja $x:\left[-\frac{\pi}{2},\frac{\pi}{2}\right]\to[-a,a]$ definida por $x=g(t)=a\sin t$. Então, $t=\arcsin\left(\frac{x}{a}\right)$ e $g'(t)=a\cos t$, ou seja, $dx=g'(t)\,dt=a\cos t\;dt$ e desta forma,

$$\begin{aligned}\int \sqrt{a^2-x^2}\,dx &= \int \sqrt{a^2-a^2\sin^2 t}\times a\cos t\;\;dt\\ &= \int \sqrt{a^2(1-sin^2 t)}a\cos t\;\;dt\\ &= \int a^2\cos^2 tdt = a^2\int \cos^2 t\;\;dt\end{aligned}$$

Da formula da duplicação do ângulo, $\cos(2t)=\cos^2 t-\sin^2 t$, vem que $\cos^2 t=\frac{1}{2}\left(1+\cos(2t)\right)$ e então,

$$\begin{aligned}a^2\int\cos^2 t\;dt &= a^2\int\frac{1}{2}\left(1+\cos(2t)\right)\;dt\\ &= \frac{a^2}{2}\int 1+\cos(2t)\;dt\\ &= \frac{a^2}{2}\left(t+\frac{\sin(2t)}{2}\right)\\ &= \frac{a^2}{2}\left(t+\frac{2\sin t\cos t}{2}\right)\\ &= \frac{a^2}{2}\left(t+\sin t\cos t\right)+C,\;\; C\in\mathbb{R}\end{aligned}$$

Como $t=\arcsin\left(\frac{x}{a}\right)$, temos

$$\frac{a^2}{2}\left(t+\sin t\cos t\right)=\frac{a^2}{2}\left(\arcsin\left(\frac{x}{a}\right)+\sin\left(\arcsin\left(\frac{x}{a}\right)\right)\cos\left(\arcsin\left(\frac{x}{a}\right)\right)\right)$$

De $\cos\left(\arcsin\left(\frac{x}{a}\right)\right)=\cos t$ e de $x=a\sin t$ vem que $\cos t=\sqrt{1-\left(\frac{x}{a}\right)^2}$ e, então,

$$\begin{aligned}&\frac{a^2}{2}\left(\arcsin\left(\frac{x}{a}\right)+\sin\left(\arcsin\left(\frac{x}{a}\right)\right)\cos\left(\arcsin\left(\frac{x}{a}\right)\right)\right)=\\ =\;&\frac{a^2}{2}\left(\arcsin\left(\frac{x}{a}\right)+\frac{x}{a}\sqrt{1-\left(\frac{x}{a}\right)^2}\right)=\\ =\;&\frac{a^2}{2}\left(\arcsin\left(\frac{x}{a}\right)+\frac{x}{a^2}\sqrt{a^2-x^2}\right)=\\ =\;&\frac{a^2}{2}\arcsin\left(\frac{x}{a}\right)+\frac{x}{2}\sqrt{a^2-x^2}+C,\quad C\in\mathbb{R}\end{aligned}$$

em $I=[-a,a]$ [4].

Capítulo 2

Cálculo Integral

Historicamente, as ideias que estiveram na origem do Cálculo Integral precederam de muitos séculos o despontar do Cálculo Diferencial.

Os primeiros problemas que surgiram relacionados com integrais foram problemas de quadratura. Os antigos geómetras estudavam as áreas de figuras planas e relacionavam-nas com a área do quadrado por ser essa a figura plana tida como mais simples. Desta forma, procuravam encontrar um quadrado que tivesse área igual à da figura em estudo. As quadraturas que mais fascinavam os geómetras eram as curvilíneas, como o círculo, ou figuras limitadas por arcos de outras curvas. Hipócrates de Chios, (470 - 410 a. C.) foi pioneiro, em 440 a.C., no estudo das lúnulas - regiões que se assemelham com a lua no seu quarto-crescente [5]. Por essa altura, outro matemático, procurou encontrar a quadratura

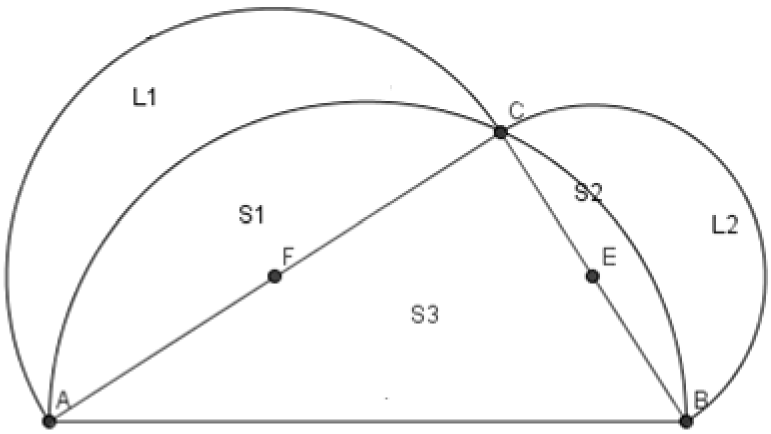


Figura 2.1: Quadratura das Lúnulas de Hipócrates de Chios

do círculo através de uma sequência infinita de polígonos regulares inscritos: primeiro um quadrado, depois um octógono, em seguida um hexadecágono, e assim por diante. Havia um problema: essa sequência nunca poderia ser concluída. Foi, no entanto, uma ideia genial que deu origem ao método da exaustão[5].

Eudoxos de Cnidos (408 - 355 a. C.) parece ter sido o criador do Método da Exaustão. Em termos intuitivos este método consiste em aproximar uma dada figura ou sólido geométrico cuja medida (comprimento, área ou volume) se pretende determinar, por figuras inscritas ou circunscritas de medidas conhecidas, tomando-se depois o limite destas medidas para a medida da figura dada. Arquimedes de Siracusa (287 - 212 a. C.), o maior génio matemático da Grécia, obteve e demonstrou diversos resultados notáveis, usando também o referido Método da Exaustão.

No entanto, uma teoria de integração satisfazendo os modernos padrões de rigor não era possível sem estarem definidos de forma adequada os fundamentos da Análise Real. Principalmente por obra de Cauchy (1789 - 1857), Riemann (1826 - 1866), Lebesgue, Radon, Fréchet... (séc. XX) foram elaboradas diversas teorias baseadas em definições rigorosas do conceito de integral[4].

A atual notação de integral que usámos no Capítulo 1, $\int f(x)\, dx$, foi introduzida pelo matemático e filósofo alemão Gottfried Wilhelm Leibniz (1646 - 1716) no final do século XVII [15].

De ora em diante, seja $I = [a, b]$ um intervalo limitado e fechado de $\mathbb{R}$, com mais de um ponto e seja f uma função definida e limitada em I.

Teorema Fundamental do Cálculo Integral

Teorema 2.1. *[15] Seja f uma função integrável em $[a, b] \subseteq \mathbb{R}$. Então a função $F : [a, b] \to \mathbb{R}$ definida por*

$$F(x) = \int_a^x f(t)\, dt$$

é contínua em $[a, b]$. Também se f for contínua em $[a, b]$, F é diferenciável em $x_0 \in [a, b]$ e tem-se

$$F'(x_0) = f(x_0).$$

O Teorema Fundamental do Cálculo é assim chamado porque estabelece uma relação entre os dois ramos do Cálculo: o cálculo diferencial e o cálculo integral. O cálculo diferencial surgiu do problema da tangente ao passo que o cálculo integral surgiu do problema do cálculo da área. Aparentemente, estes dois problemas não tinham relação entre si. O mentor de Isaac Newton (1642 - 1727) na Universidade de Cambridge, Isaac Barrow (1630 - 1677) descobriu que esses dois problemas estão intimamente relacionados. De facto, Barrow concluiu que a diferenciação e a integração são processos inversos. O Teorema Fundamental do Cálculo estabelece a exata relação inversa entre a derivada e o integral. Newton e Leibniz exploraram esta relação e concluiram que o Teorema Fundamental permite calcular áreas e integrais muito facilmente sem terem que os calcular como sendo o limite das somas de Riemann [15] [7].

Quando, em 1635, o matemático Francês Gilles de Roberval (1602 - 1675) pela primeira vez determinou a área "abaixo das curvas" do seno e do cosseno este era um problema desafiante que requeria uma certa dose de ingenuidade. Se não tivéssemos o benefício do Teorema Fundamental do Cálculo teríamos que calcular um difícil limite de somas. Tal cálculo foi ainda mais difícil para Roberval porque a notação para os limites ainda não tinha sido inventada em 1635. Contudo, nos anos sessenta e setenta do século XVII quando o Teorema Fundamental do Cálculo foi descoberto por Barrow e explorado por Newton e Leibniz, esses problemas tornaram-se muito simples [15].

A Fórmula de Barrow nas várias notações usuais [12], escreve-se:

$$\int_a^b F'(t)dt = [F(x)]_a^b = F(x)\,\Big|_a^b = F(b) - F(a).$$

Um exemplo interessante dos conceitos de primitiva e derivada é a mecânica newtoniana. Como é sabido, o conceito matemático de derivada está relacionado com o conceito físico de velocidade. A equação do movimento

$$s = s\,(t)$$

permite calcular o espaço percorrido por uma partícula em função do tempo t, contado a partir de um certo instante inicial.

Derivando a equação $s = s\,(t)$ em ordem a t temos,

$$v = s'\,(t) = \frac{ds}{dt}.$$

Podemos agora considerar o problema inverso. Dada uma equação que descreve a velocidade de um corpo $v = f(t)$, procurar a equação do movimento desse corpo.

Matematicamente, o problema pode ser apresentado da forma seguinte: dada uma função f, determinar uma função F cuja derivada seja f.

Este problema nem sempre tem solução, isto é, pode ser impossível pois há funções para as quais não é possível encontrar uma primitiva. Quando o problema é possível é, no entanto, indeterminado pois, conforme já foi visto na secção 1.1, se $F(x)$ é primitiva de $f(x)$ em $I \subseteq \mathbb{R}$ então, $f(x) + C$ é também primitiva de $f(x)$ em I, para qualquer número real C e para qualquer $x \in I$.

Num movimento retilíneo uniformemente acelerado, o integral da aceleração a em função do tempo t, $\int a\ dt$, dá-nos a velocidade do móvel em função do tempo: $v = \int a\ dt = v_0 + at$ onde v_0 é a constante de primitivação que corresponde à velocidade inicial do móvel. Por sua vez, o integral da velocidade em função do tempo $\int v\,dt$ dá-nos a posição do móvel em função do tempo: $s = \int v\ dt$.

Sendo $s = \int v\ dt$ e sendo $v = \int a\ dt$ vem:

$$s = \int v\ dt = \int (v_0 + at)\ dt = s_0 + v_0 t + \frac{1}{2} a\, t^2$$

onde s_0 é a constante de integração que corresponde à posição inicial do móvel.

2.1 Integral definido

Definição 2.1. Seja f uma função, real de variável real, definida num intervalo $[a, b] \subseteq \mathbb{R}$. Chama-se *partição* P desse intervalo a qualquer decomposição de $[a, b]$, em n subintervalos da forma $[x_{i-1}, x_i]$ tais que: $a = x_0 < x_1 < \cdots < x_n = b$

Uma *partição* P do intervalo $[a, b]$ divide $[a, b]$ em n intervalos

$$[x_0, x_1], [x_1, x_2], \ldots, [x_{n-1}, x_n]$$

A amplitude de cada intervalo $[x_{i-1}, x_i]$ é notada por $\Delta x_i = x_i - x_{i-1}$, com $i = 1, 2, \ldots, n$. As amplitudes dos intervalos não são necessariamente iguais. Este processo divide a região situada entre o eixo Ox e o gráfico da função em n faixas como podemos ver adiante.

Definição 2.2. Seja f uma função, real de variável real, definida num intervalo $[a, b]$ e P uma partição desse intervalo. Chama-se *Soma de Riemann* de f em relação à partição P, a toda a expressão da forma:

$$\sum_{i=1}^{n} f(w_i)\Delta x_i$$

onde w_i é um valor qualquer do intervalo $[x_{i-1}, x_i]$, com $i = 1, 2, \ldots, n$.

Denotamos por m_i e M_i respetivamente o ínfimo e o supremo de $f(x)$ no intervalo $[x_{i-1}, x_i]$.

$$m_i = \inf f(x) \quad \text{e} \quad M_i = \sup f(x)$$

onde $x \in [x_{i-1}, x_i]$ e $i = 1, 2, \ldots, n$ [12].

Definição 2.3. [12]

Chamamos *Soma Inferior de Darboux* da função f relativa à partição P ao número

$$s(f, P) = \sum_{i=1}^{n} m_i(x_i - x_{i-1})$$

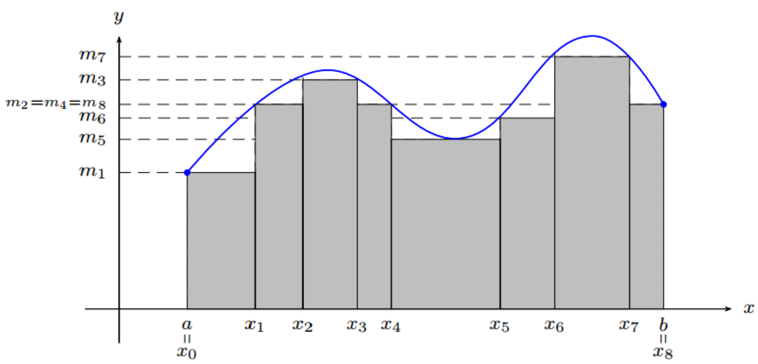


Figura 2.2: [2] Interpretação geométrica das *Somas Inferiores de Darboux* de uma função $f : [a, b] \to \mathbb{R}$

Do mesmo modo, chamamos *Soma Superior de Darboux* da função f relativa à partição P ao número

$$S(f, P) = \sum_{i=1}^{n} M_i(x_i - x_{i-1})$$

Os valores das Somas Inferiores e Superiores de Darboux correspondem à soma das áreas dos retângulos, como ilustrado nas figuras 2.2 e 2.3, respetivamente.

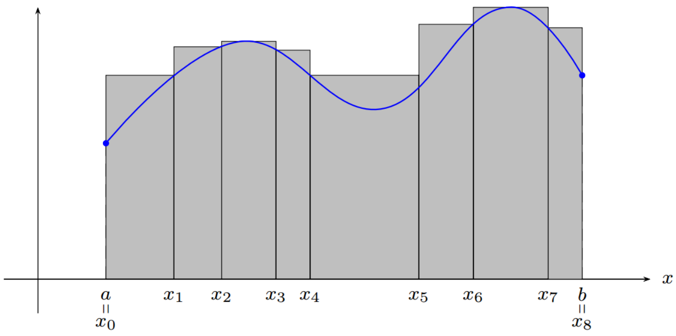


Figura 2.3: [2] Interpretação geométrica das *Somas Superiores de Darboux* de uma função $f : [a, b] \to \mathbb{R}$

Definição 2.4. Chama-se *Integral Inferior de Darboux* de f em $[a, b]$ ao número

$$S = \sup_P s(f, P)$$

Este valor é o supremo relativo a todas as partições P de $[a, b]$.

Análogamente,

Definição 2.5. Chama-se *Integral Superior de Darboux* de f em $[a, b]$ ao número

$$s = \inf_P s(f, P)$$

Se f é limitada em $[a, b]$, existem dois números m e M tais que $m \leqslant f(x) \leqslant M$ em $[a, b]$ e, portanto,

$$s(f, P) = \sum_{i=1}^{n} m_i(x_i - x_{i-1}) \geqslant \sum_{i=1}^{n} m(x_i - x_{i-1}) = m(x_n - x_0) = m(b - a).$$

Analogamente se prova que $S(f, P) \leqslant M(b - a)$ e podemos escrever,

$$m(b - a) \leqslant s(f, P) \leqslant S(f, P) \leqslant M(b - a).$$

Definição 2.6. Quando os integrais inferior e superior de Darboux em $[a, b]$ coincidem, a função f diz-se integrável à Riemann em $[a, b]$. Ao valor coincidente desses integrais chama-se *Integral de Riemann* de f em $[a, b]$ e representa-se por

$$\int_a^b f(x)\, dx.$$

A a e a b chamamos limites de integração: a é o limite inferior do integral; b é o limite superior do integral.

Geometricamente, caso f seja uma função positiva, o Integral de Riemann define a área exata da região entre o eixo Ox e o gráfico da função.

Sempre que utilizamos um intervalo $[a, b]$, supomos que $a < b$. Mas a definição anterior pode ser estendida ao caso $a > b$:

$$\int_a^b f(x)\,dx = -\int_b^a f(x)\,dx.$$

Como consequência imediata temos o resultado seguinte[8]:

$$\int_a^a f(x)\,dx = 0$$

As três propriedades apresentadas na tabela seguinte são válidas para quaisquer a e b [15].

Propriedades dos Integrais
1. $\int_a^b c\,dx = c(b-a)$, onde c é uma constante
2. $\int_a^b [f(x)+g(x)]\,dx = \int_a^b f(x)\,dx + \int_a^b g(x)\,dx$
3. $\int_a^b cf(x)\,dx = c\int_a^b f(x)\,dx$, onde c é uma constante

Tabela 2.1.1: Propriedades dos Integrais

A **Propriedade 1** diz que o integral de uma função constante $f(x) = c$ é igual ao produto da constante pela amplitude do intervalo. Em particular, quando $c > 0$ e $a < b$ o valor do integral corresponde à área de um retângulo de medidas c e $b - a$.

A **Propriedade 2** diz que o integral da soma de funções é igual à soma dos integrais das parcelas e decorre do facto de o limite de uma soma ser igual à soma dos limites das parcelas.

A **Propriedade 3** decorre do facto de o limite de um produto ser igual ao produto dos limites.

As Propriedades 2 e 3 também se podem escrever na forma da proposição seguinte:

Proposição 2.1. ***(Linearidade do integral)*** *Sejam f e g duas funções contínuas um intervalo I de $\mathbb{R}$. Para todos $\lambda, \mu \in \mathbb{R}$ e para todos $a, b \in I$ tem-se:*

$$\int_a^b (\lambda f(x) + \mu g(x))\ dx = \lambda \int_a^b f(x)dx + \mu \int_a^b g(x)\ dx.$$

Proposição 2.2. *[15]* ***(Relação de Chasles)*** *Sejam a, b e c três elementos de um intervalo I de $\mathbb{R}$ e f uma função contínua em I. Tem-se que:*

$$\int_a^b f(x)\ dx = \int_a^c f(x)\ dx + \int_c^b f(x)\ dx.$$

A Proposição 2.2 prova-se, tomando F uma primitiva de f em I,

$$\int_a^b f(x)\ dx = F(b) - F(a) = F(b) - F(a) - F(c) + F(c) =$$

$$= (F(c) - F(a)) + (F(b) - F(c)) = \int_a^c f(x)\ dx + \int_c^b f(x)\ dx.$$

Propriedades Comparativas dos Integrais
4. Se $f(x) \geqslant 0$ para $a \leqslant x \leqslant b$, então $\int_a^b f(x)dx \geqslant 0$
5. Se $f(x) \geqslant g(x)$ para $a \leqslant x \leqslant b$, então $\int_a^b f(x)\ dx \geqslant \int_a^a g(x)\ dx$
6. Se $m \leqslant f(x) \leqslant M$ para $a \leqslant x \leqslant b$, então $m(b-a) \leqslant \int_a^b f(x)\ dx \leqslant M(b-a)$

Tabela 2.1.2: Propriedades Comparativas dos Integrais

As três propriedades anteriores são válidas apenas quando $a \leqslant b$ [15].

2.2 Áreas de regiões planas

O integral de uma função $f : [a, b] \subseteq \mathbb{R} \to \mathbb{R}$ corresponde à área da parte do plano xOy limitado pelo gráfico de f, pelo eixo Ox e pelas retas verticais $x = a$ e $x = b$. [2]

Se $f \geqslant 0$, para $x \in [a, b]$, a situação é ilustrada na Figura 2.4,

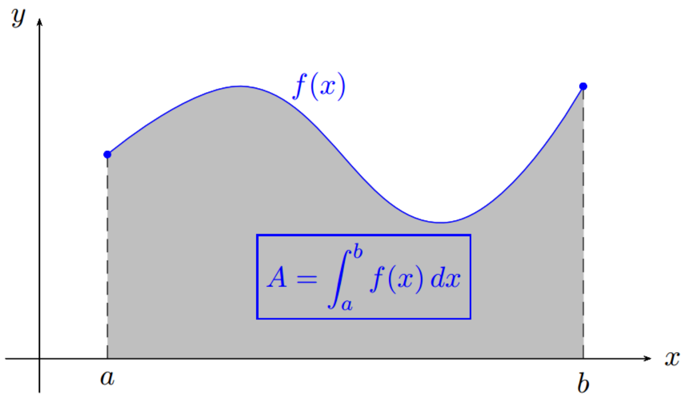


Figura 2.4: [2] Caso em que f é positiva em $[a, b]$

e se $f < 0$, para $x \in [a, b]$, a situação é aquela que é ilustrada na Figura 2.5.

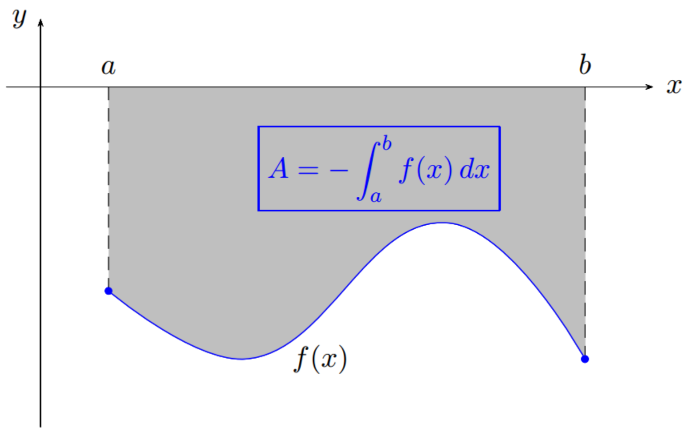


Figura 2.5: [2] Caso em que f é negativa em $[a, b]$

Para os casos em que a mesma função é positiva numa parte de $[a, b]$ e negativa noutra parte de $[a, b]$, podemos calcular os integrais em separado:

O caso em que $f : [a, b] \to \mathbb{R}$ é uma função integrável, existindo $c \in\,]a, b[$ tal que $f(x) \geqslant 0$ para qualquer $x \in [a, c]$ e $f(x) \leqslant 0$ para qualquer $x \in [c, b]$ é ilustrado na figura seguinte:

A próxima figura ilustra o caso em que a área a determinar se situa entre duas funções f e g tais que $f(x) \geqslant g(x)$ para qualquer $x \in [a, b]$

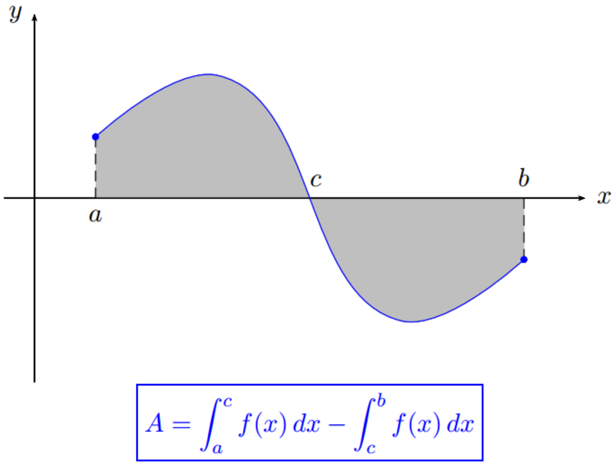


Figura 2.6: [2] Caso em que f não tem sinal constante em $[a, b]$

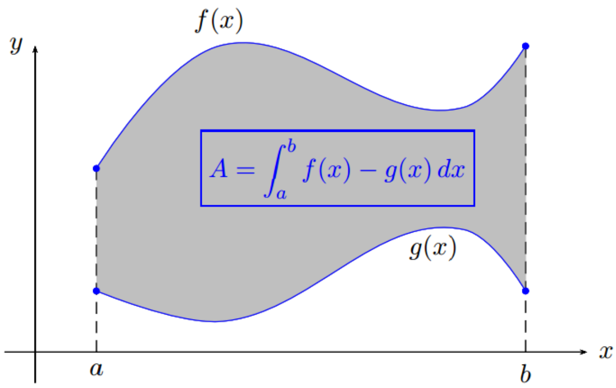
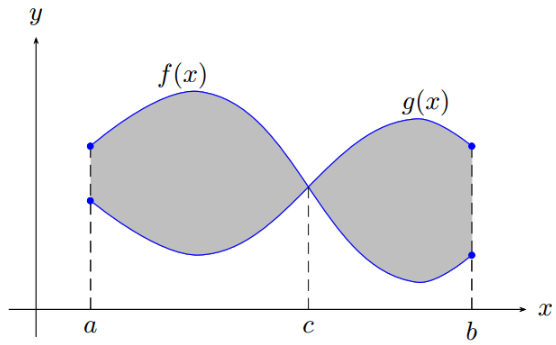


Figura 2.7: [2] Caso em que $f(x) > g(x)$ para qualquer $x \in [a, b]$

Para os casos em que a função f é superior a g para determinados subconjuntos de $[a, b]$ e inferior para outros subintervalos de $[a, b]$, podemos calcular os integrais em separado como ilustra a figura que se segue:



$$\boxed{A = \int_a^c f(x) - g(x)\,dx + \int_c^b g(x) - f(x)\,dx}$$

Figura 2.8: [2] Caso em que $f(x) \geqslant g(x)$ para qualquer $x \in [a, c]$ e $f(x) \leqslant g(x)$ para qualquer $x \in [c, b]$

Exemplo 2.1. Calcular a área da região plana limitada pela reta de equação $y = x + 2$ e pela parábola de equação $y = x^2$:

Comecemos por calcular os pontos de intersecção das duas curvas

$$\left\{\begin{array}{lcl} y & = & x+2 \\ y & = & x^2 \end{array}\right. \Leftrightarrow \left\{\begin{array}{lcl} x^2 & = & x+2 \\ y & = & x^2 \end{array}\right. \Leftrightarrow \left\{\begin{array}{lcl} x & = & -1 \;\vee\; x \;=\; 2 \\ y & = & x^2 \end{array}\right.$$

Se $x = -1$ vem $y = 1$ e se $x = 2$ vem $y = 4$, sendo os pontos de intersecção $(2, 4)$ e $(-1, 1)$.

Representemos geometricamente a região do plano da qual queremos calcular a área:

Assim sendo, a área pretendida é dada por:

$$\begin{aligned} \int_{-1}^{2} \left(x + 2 - x^2\right)\,dx &= \left[\frac{x^2}{2} + 2x - \frac{x^3}{3}\right]_{-1}^{2} \\ &= \frac{4}{2} + 4 - \frac{8}{3} - \frac{1}{2} + 2 - \frac{1}{3} \\ &= \frac{9}{2} \end{aligned}$$

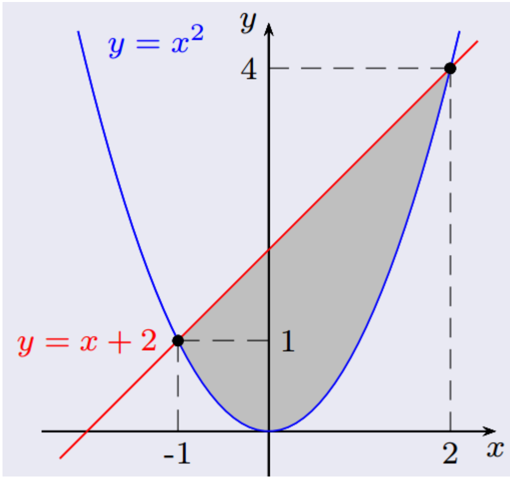


Figura 2.9: [2] Representação gráfica de $y = x + 2$ e $y = x^2$

Exemplo 2.2. [7] Sejam $f(x) = 2x$, $g(x) = \dfrac{1}{x}$ e $h(x) = \dfrac{x}{x^2+1}$.
Calcular a medida da área da região do plano delimitada pelos gráficos de f, g e h e pela reta de equação $x = 1$.

Comecemos por representar geometricamente a parte do gráfico das três funções f, g e h e da reta de equação $x = 1$ correspondente à área que pretendemos calcular.

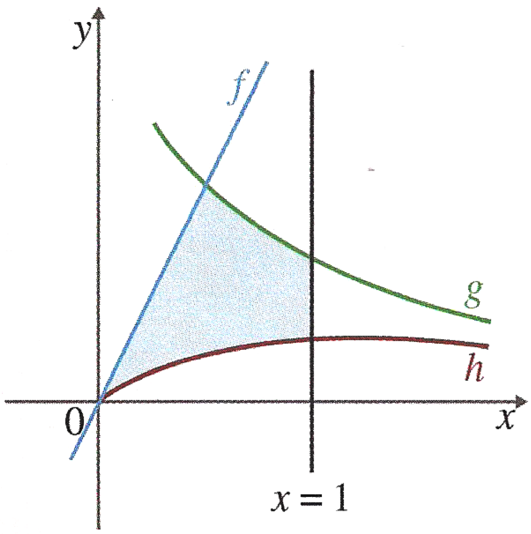


Figura 2.10: [7] Representação gráfica de $f(x) = 2x$, $g(x) = \dfrac{1}{x}$, $h(x) = \dfrac{x}{x^2+1}$ e $x = 1$

Seguidamente calculemos os pontos de interseção das funções f e h.

$f(x) = h(x) \Leftrightarrow 2x = \dfrac{x}{x^2+1} \Leftrightarrow x\left(\dfrac{2x^2+1}{x^2+1}\right) = 0 \Leftrightarrow \big(x = 0 \vee 2x^2+1 = 0\big) \wedge x^2+1 \neq 0 \Leftrightarrow x = 0$

Ora, quando $x = 0$ vem $f(0) = h(0) = 0$ e o ponto de interseção dos gráficos das funções f e h é a origem do referencial.

Calculemos, agora, os pontos de interseção das funções f e g.

$f(x) = g(x) \Leftrightarrow 2x = \dfrac{1}{x} \Leftrightarrow x\dfrac{1-2x^2}{x} = 0 \Leftrightarrow 1-2x^2 = 0 \wedge x \neq 0 \Leftrightarrow x = -\dfrac{\sqrt{2}}{2} \vee x = \dfrac{\sqrt{2}}{2}$

Para calcular a área pretendida interessa-nos apenas a interseção de f e g quando $x = \dfrac{\sqrt{2}}{2}$, ou seja, o ponto de coordenadas $\left(\dfrac{\sqrt{2}}{2}, \sqrt{2}\right)$

Para o cálculo da área que pretendemos há necessidade de dividir a figura pela reta de equação $x = \dfrac{\sqrt{2}}{2}$, como se pode ver na Figura 2.11:

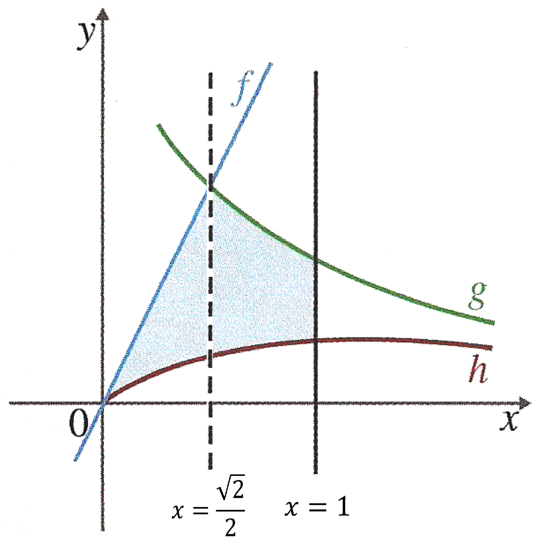


Figura 2.11: [7] Divisão da área pretendida pela reta de equação de $x = \dfrac{\sqrt{2}}{2}$

Portanto, a medida da área pedida pode ser dada por:

$$\begin{aligned}
& \int_0^{\frac{\sqrt{2}}{2}} (f(x) - h(x)) \; dx + \int_{\frac{\sqrt{2}}{2}}^1 (g(x) - h(x)) \; dx = \\
= \; & \int_0^{\frac{\sqrt{2}}{2}} \left(2x - \frac{x}{x^2+1}\right) \; dx + \int_{\frac{\sqrt{2}}{2}}^1 \left(\frac{1}{x} - \frac{x}{x^2+1}\right) \; dx = \\
= \; & \int_0^{\frac{\sqrt{2}}{2}} 2x \; dx - \int_0^{\frac{\sqrt{2}}{2}} \frac{x}{x^2+1} \; dx + \int_{\frac{\sqrt{2}}{2}}^1 \frac{1}{x} \; dx - \int_{\frac{\sqrt{2}}{2}}^1 \frac{x}{x^2+1} \; dx = \\
= \; & 2\left[\frac{x^2}{2}\right]_0^{\frac{\sqrt{2}}{2}} - \frac{1}{2}\left[\ln\left|x^2+1\right|\right]_0^{\frac{\sqrt{2}}{2}} + \left[\ln|x|\right]_{\frac{\sqrt{2}}{2}}^1 - \frac{1}{2}\left[\ln\left|x^2+1\right|\right]_{\frac{\sqrt{2}}{2}}^1 = \\
= \; & 2\left(\frac{1}{4} - 0\right) - \frac{1}{2}\left(\ln\left(\frac{3}{2}\right) - 0\right) + \left(0 - \ln\left(\frac{\sqrt{2}}{2}\right)\right) - \frac{1}{2}\left(\ln(2) - \ln\left(\frac{3}{2}\right)\right) = \\
= \; & \frac{1}{2} - \frac{1}{2}\ln\left(\frac{3}{2}\right) - \frac{1}{2}\ln\left(\frac{1}{2}\right) - \frac{1}{2}\ln\left(\frac{4}{3}\right) = \\
= \; & \frac{1}{2} - \frac{1}{2}\ln\left(\frac{3}{2} \times \frac{1}{2} \times \frac{4}{3}\right) = \\
= \; & \frac{1}{2} - \ln(1) = \frac{1}{2}.
\end{aligned}$$

Capítulo 3

Construção de um exercício parametrizado

3.1 Introdução

Neste capítulo faremos uma descrição dos procedimentos inerentes à construção de um exercício parametrizado. Um exercício parametrizado é, na verdade, um conjunto ou classe de exercícios que incidem sobre os mesmos conceitos matemáticos e sobre o qual recaem os mesmos objetivos pedagógicos e didáticos. Sempre que se concretizam os parâmetros obtém-se um exercício concreto.

Neste texto, na ótica do utilizador, cada exercício está estruturado em três partes, a saber:

- O enunciado do problema
- Quatro afirmações das quais apenas uma é verdadeira.
- Resolução do problema que pode ser consultada independentemente de se ter respondido ou não à questão.

No ponto seguinte apresentamos estrutura da construção de um exercício.

3.2 A construção de um exercício

A construção dos exercícios desta monografia requer o *software* matemático *SageMath* de código aberto que engloba e faz uso de um grande número de pacotes pré-existentes como Maxima, GAP, Pari/GP e *softwares* de renderização de imagens. O nome *SageMath* é um acrónimo em inglês para Sistema Algébrico e Geométrico de Experimentações [14].

A biblioteca de software *open source* MEGUA funciona sobre o sistema de computação para matemática *SageMath* e permite a criação de arquivos de exercícios escritos na linguagem tipográfica LaTeX. A linguagem de programação usada é *Python* com acesso às bibliotecas do *SageMath*. O nome "MEGUA"designa uma marca registada da Universidade de Aveiro [6].

A estrutura base de um exercício parametrizado criado com o MEGUA tem a forma que se apresenta a seguir.

```
meg.save( r’’ ‘
%summary Texto Secção; Texto Subsecção; Texto Subsubsecção
Palavras-chave:
Autores:
Ano:
Propósito didático:
%problem [Objetivo geral do exercício]
----- Enunciado do Exercício -----
%ANSWER
%(Opções de escolha múltipla)
<multiplechoice>
<choice> opção correta </choice>
<choice> opção errada 1 </choice>
<choice> opção errada 2 </choice>
<choice> opção errada 3 </choice>
</multiplechoice>
% Resolução do exercício (em LaTeX)
```

```
class [nome/número do exercício]
def make$_$random(s,edict=None):
Definição das variáveis e dos parâmetros. Definição dos conjuntos em que cada parâmetr
Definição da função a integrar
s.f1 =)
Definição da resposta certa
s.resposta0 =
Definição das respostas erradas e erro de cálculo respetivo
s.re1 = Erro de s.re1
s.re2 = Erro de s.re2
s.re3 = Erro de s.re3
’’ ‘)
```

Em *summary* são colocadas por ordem decrescente as secções às quais pertence o exercício em causa de acordo com a definição prévia do curso que se está a construir. Depois é identificado o exercício através de palavras-chave, os autores, o ano e o propósito didático do exercício. O enunciado do exercício, bem como a sua resolução são escritas em LaTeX. As opções de escolha múltipla enquadradas no comando *multiplechoice* remetem para as respostas certa e erradas previamente construídas. A resposta certa é, habitualmente, produzida de forma automática por recurso, neste caso particular, à função *integrate*. As respostas erradas (também chamadas distratores) são produzidas pelo programador em alguns casos, também com recurso a funções automáticas como é o caso da função *derivate*. Podem ser produzidas mais de três respostas erradas de entre as quais são escolhidas aleatoriamente três erradas para incluir no grupo de respostas. Estas respostas erradas são construídas a partir dos erros mais frequentes dos alunos para que, desta forma, além de avaliar os conhecimentos e competências adquiridos pelos alunos, o professor possa também identificar concretamente os conhecimentos e competências não adquiridos pelos mesmos.

O erro cometido propositadamente na resolução de cada exercício com vista à obtenção de uma resposta errada foi coerente ao longo de cada exercício, isto é, cometemos o mesmo erro em todos os momentos em que ele pode ocorrer num mesmo exercício. Pretendemos desta forma perceber cabalmente se o aluno de facto não domina determinado pormenor. Uma resposta errada em que surja a mesma situação várias vezes mas em que o aluno errou num momento e acertou noutro pode indicar que ocorreu apenas um erro de distração.

A escolha de uma resposta errada por parte do aluno, permite ao professor intuir com grande grau de confiança que o aluno não adquiriu determinado conhecimento ou determinada competência, tendo assim a função de redefinir o trabalho a ser efetuado futuramente com vista a cercear tanto quanto possível os erros identificados.

Terminado o exercício, o seu enunciado, as suas respostas (certa e erradas) e a sua resolução podem ser visualizadas numa página *HTML*. As respostas aparecem ordenadas (Resposta certa seguida pelas três erradas de acordo com a numeração atribuída pelo programador). Esta visualização ordenada ocorre apenas na ótica do programador pois o utilizador verá as respostas apresentadas de forma aleatória. O sistema devolve aleatoriamente uma das chaves (`ekey`) possíveis. No entanto, pode o utilizador dar ordem de publicação de uma chave em concreto, fazendo para tal `ekey =` n.º da chave pretendido e pertencente ao intervalo previamente definido. Pode ainda o utilizador visualizar todas as chaves possíveis dentro do intervalo de chaves previamente definido.

Finalizado o exercício, o mesmo é exportado para o Sistema "SIACUA" (Sistema Interativo de Aprendizagem por Computador da Universidade de Aveiro). O SIACUA foi criado por docentes da Universidade de Aveiro e pode ser acedido em `http://siacua.web.ua.pt/` [13].

Apresentamos em seguida uma concretização de um exercício parametrizado. O caso apresentado corresponde ao exercício `E26A36_Antidiferentiation_Parts_026_siacua` constante no **Apêndice A** que contém a lista de exercícios por nós produzidos.

Com este exercício pretendemos que os alunos identifiquem uma família de primitivas da função definida por:

$$f(x) = a_1 \times \ln(a_2 \times x)$$

```
meg.save(r'''
%summary Primitivas; Primitivas Por Partes


Palavras-chave: Primitiva; antiderivada.
Autores: João Silva
```

```
Ano: 2017
Propósito didático: Primitivação por partes

%problem Calcular primitivas usando o método e primitivação por partes:a ln(b x)

Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser'

<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>

%ANSWER

Uma primitiva desta função pode ser obtida utilizando a regra de primitivação por part
<p>
Assim,
\begin{eqnarray*}
\int f1 \, dx
&=& a1 \, \int 1 \, \times \, \ln(a2 x)\, dx \\
&=& a1 \, \left( x \, \ln (a2 x) - \int x \frac {a2}{a2 x} \, dx \right) \\
&=& a1 \, \left( x \, \ln (a2 x) - x \right)\\
&=& a1 \, x \, \left(\ln (a2 x) - 1 \right)\\
&=&  resposta0.
\end{eqnarray*}

Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portant
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
```

```
class E26A36_Antidiferentiation_Parts_026_siacua(ExSiacua):

def make$_$random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar
s.a2 = ur.iunif_nonset(0,9,[0,1,-1]) # parametro a multiplicar
s.f1 = s.a1*ln(s.a2*x) #função a integrar
s.resposta0 =  integrate(s.f1,x)

#respostas erradas
s.re1 = diff(s.f1, x)
s.re2 =   s.a1*x*(ln(s.a2*x - 1/(s.a2)))
s.re3 =   s.a1/2 *(ln (s.a2*x))^2
''')
```

O comando `meg.save(r'''` dá início ao exercício. Em *summary* são colocadas por ordem decrescente as secções às quais pertence o exercício em causa: Primitivas; Primitivas Por Partes. Depois é identificado o exercício através de `Palavras-chave`: Primitiva; antiderivada., os `Autores`: João Silva, o `Ano`: 2017 e o `Propósito didáctico`: Primitivação por partes. O enunciado do exercício, bem como a sua resolução são escritas em LaTeX.

As opções de escolha múltipla enquadradas no comando *multiplechoice* remetem para as respostas certa e erradas previamente construídas. A resposta certa é `resposta0 + C`. As resposta erradas `re1 + C`, `re2 + C` e `re3 + C` estão definidas adiante.

Em `%ANSWER` está descrita a resposta que o aluno irá visualizar onde se inclui o cálculo da primitiva de forma faseada:

```
Uma primitiva desta função pode ser obtida utilizando a regra de primitivação
por partes.
<p>
Assim,
\begin{eqnarray*}
```

```
\int f1 \, dx
&=& a1 \, \int 1 \, \times \, \ln(a2 x)\, dx \\
&=& a1 \, \left( x \, \ln (a2 x) - \int x \frac {a2}{a2 x} \, dx \right) \\
&=& a1 \, \left( x \, \ln (a2 x) - x \right)\\
&=& a1 \, x \, \left(\ln (a2 x) - 1 \right)\\
&=&  resposta0.
\end{eqnarray*}
Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva,
portanto, qualquer função do tipo $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$
é uma primitiva de $f$.
```

Em `def make_random(s,edict=None):` define-se:

- a variável: `x=var('x');`
- os parâmetros: `s.a1 = ur.iunif_nonset(-9,9,[0,1,-1])` e `s.a2 = ur.iunif_nonset(0,9,[0,1,-1]);`
- a função integranda: `s.f1 = s.a1*ln(s.a2*x);`
- a resposta certa: `s.resposta0 = integrate(s.f1,x)` calculada com recurso à função `integrate` em *Python*;
- a resposta errada: `s.re1 = diff(s.f1, x)`, seguida da descrição do erro cometido: A resposta `s.re1` foi definida com recurso à função de derivada `diff` em *Python* onde o erro cometido é, obviamente, derivar em vez de primitivar;
- a resposta errada: `s.re2 = s.a1*x*(ln(s.a2*x - 1/(s.a2)))`, seguida da descrição do erro cometido: A resposta `s.re2` foi construída pelo programador e o erro cometido foi calcular $(\ln u)' = \frac{1}{u}$ em vez de $\frac{u'}{u}$ por generalização errada de $(\ln x)' = \frac{1}{x}$;
- a resposta errada: `s.re3 = s.a1/2 *(ln (s.a2*x))^2` seguida da descrição do erro cometido: A resposta `s.re3` foi construída pelo programador e o erro cometido foi calcular a primitiva de $\ln(a_2 x)$ como sendo a base de potência de expoente 1 mesmo sem ter o produto pela derivada da base;

Finalizado o exercício, o mesmo foi exportado para o Sistema “SIACUA” que exploramos a seguir.

3.3 Apresentação do exercício no *CoCalc*

O *CoCalc* é uma aplicação Web que permite usar o *SageMath* descrito anteriormente, bem como a biblioteca MEGUA, de forma interativa.

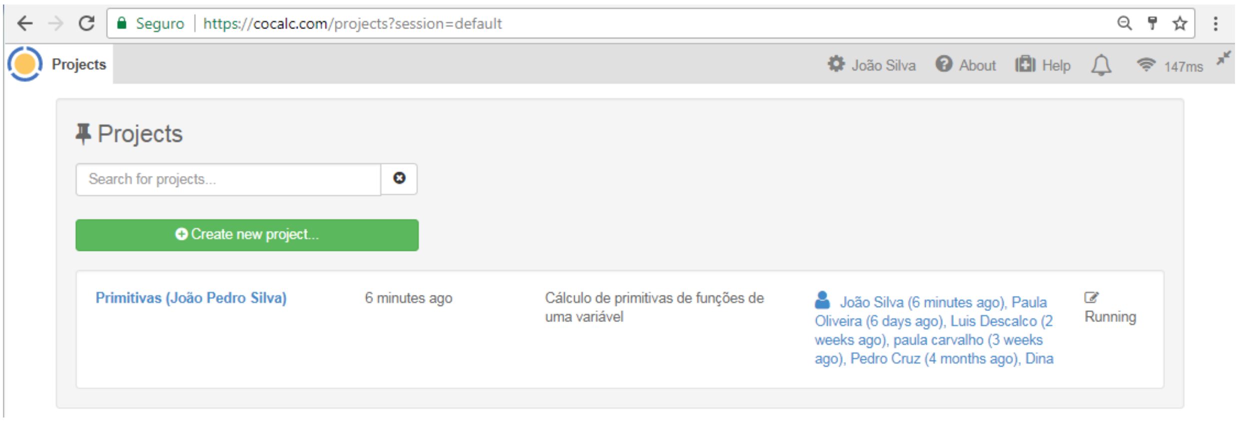


Figura 3.1: Página Inicial do *CoCalc*

Clicando em Primitivas (João Pedro Silva) podem ver-se várias pastas com conteúdos diversos mas, a pasta de trabalho é pasta *ENUNCIADOS* que se pode ver na figura 3.3:

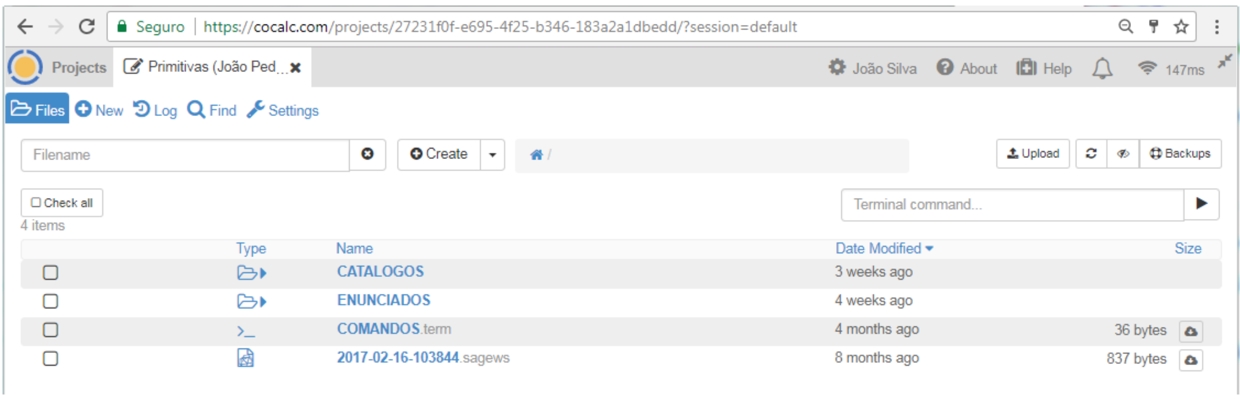


Figura 3.2: Página para escolher a pasta dos ENUNCIADOS

Clicando em ENUNCIADOS:

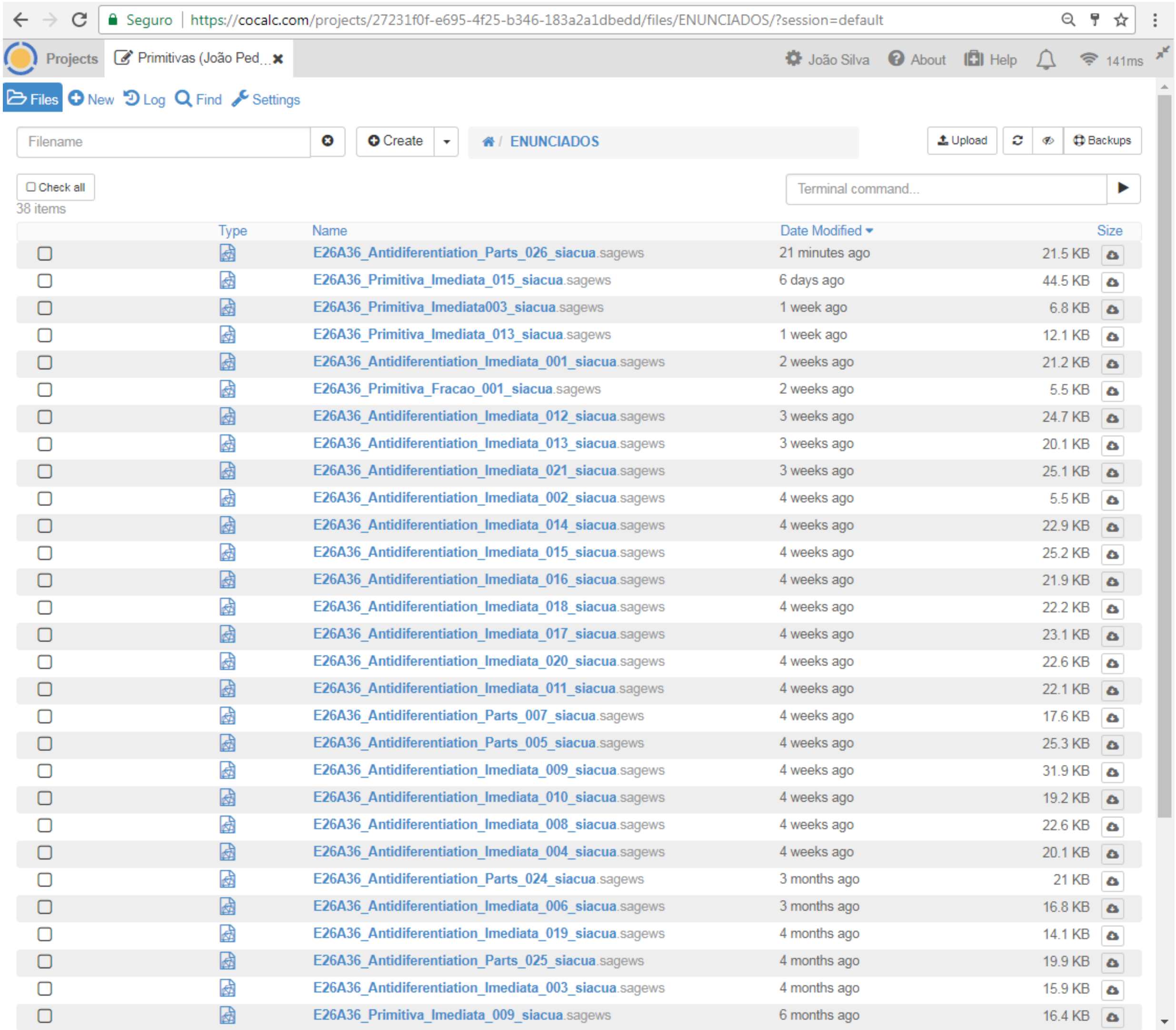


Figura 3.3: Página com os exercícios já produzidos

Escolhendo o exercício `E26A36_Antidiferentiation_Parts_026_siacua`, temos acesso a uma página extensa que se apresenta em seis imagens.

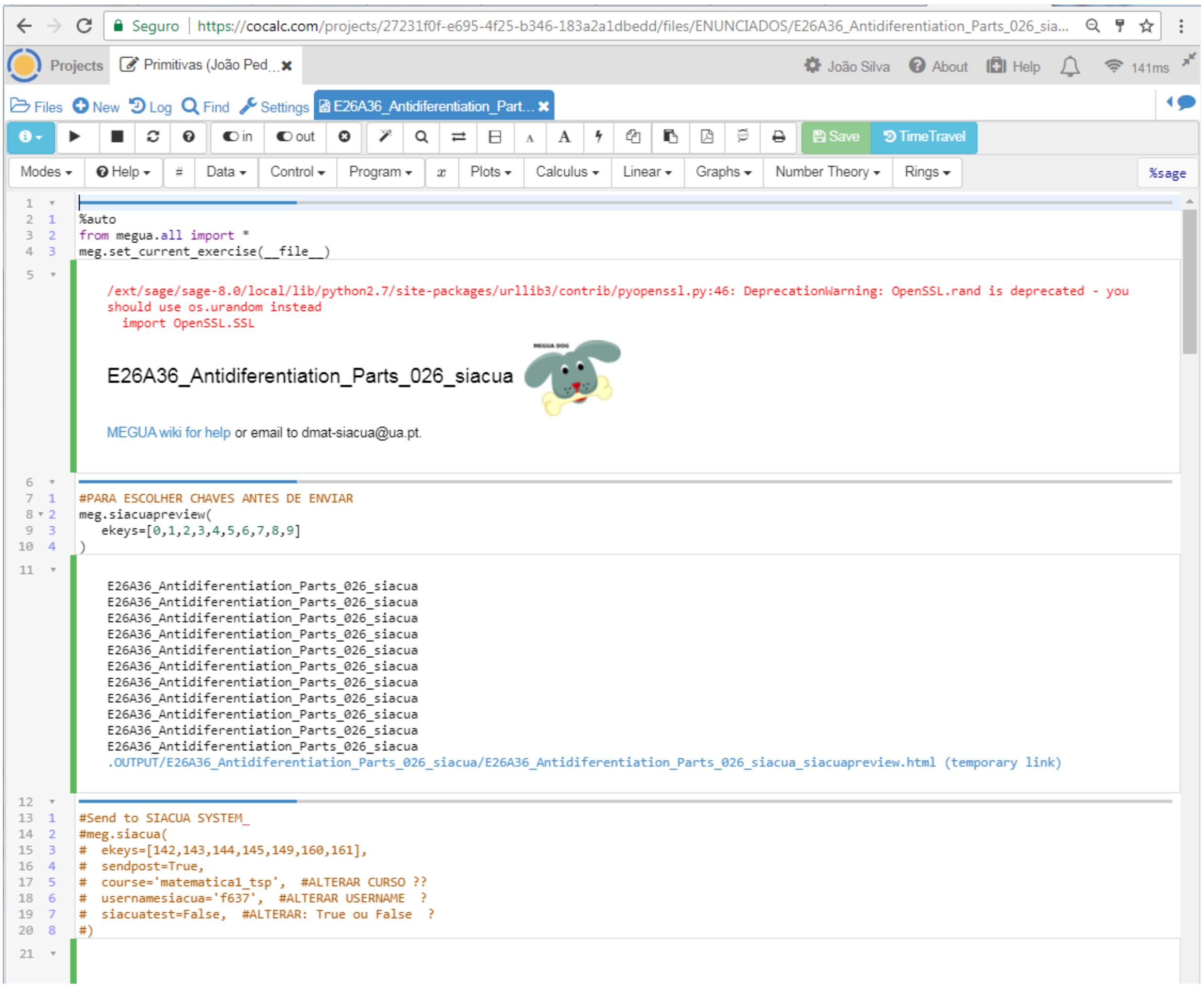


Figura 3.4: Exercício E26A36_Antidiferentiation_Parts_026_siacua (1)

A primeira célula da Figura 3.4 contém os comandos através dos quais é carregado o pacote *MEGUA*:

```
# auto
from megua.all import *
meg.set_current_exercice(__file__)
```

A segunda célula contém as chaves (`ekeys`) geradas pelo *MEGUA*:

```
#PARA ESCOLHER CHAVES ANTES DE ENVIAR
meg.siacuapreview(
```

```
ekeys=[0,1,2,3,4,5,6,7,8,9]
```

A terceira célula contém o comando (`#Send to SIACUA SYSTEM`) para exportar o exercício para o sistema *SIACUA*:

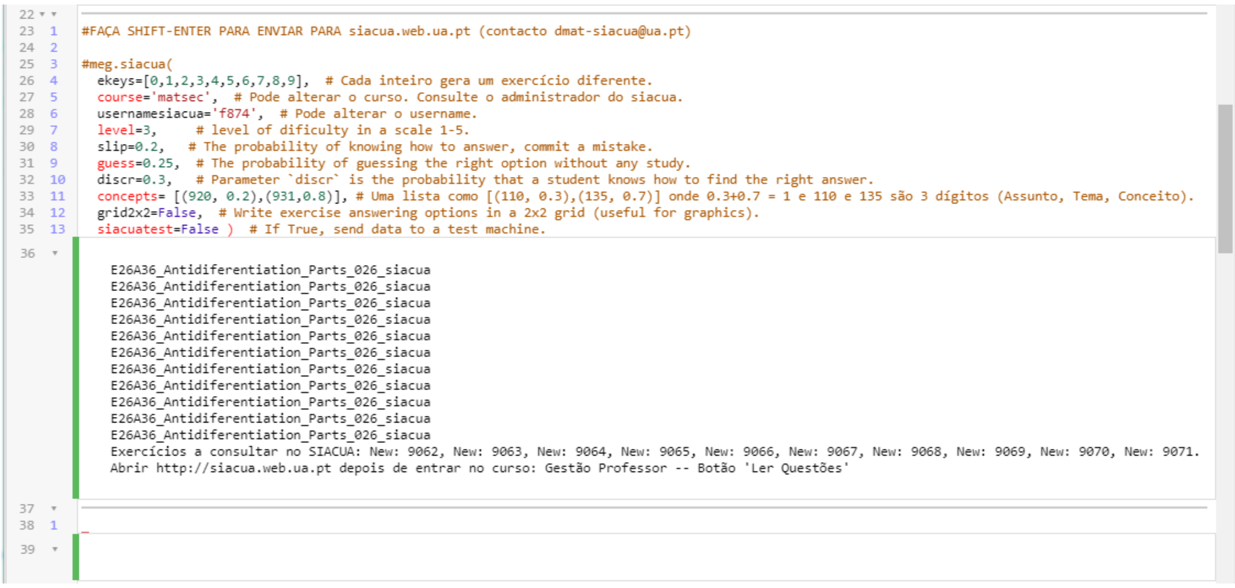
```
#Send to SIACUA SYSTEM
#meg.siacua(
#   ekeys=[142,143,144,145,149,160,161],
#   sendpost=True,
#   course='matematica1_tsp',  #ALTERAR CURSO ??
#   usernamesiacua='f637',  #ALTERAR USERNAME  ?
#   siacuatest=False,  #ALTERAR: True ou False  ?
#)
```

A imagem da Figura

refsiacua2 contém instruções tais como o nível de dificuldade, a probabilidade de acertar a resposta quando respondida de forma aleatória, a lista de conceitos contemplados no exercício, entre outras.

```
#FAÇA SHIFT-ENTER PARA ENVIAR PARA siacua.web.ua.pt (contacto dmat-siacua@ua.pt)


#meg.siacua(
ekeys=[0,1,2,3,4,5,6,7,8,9],  # Cada inteiro gera um exercício diferente.
course='matsec',  # Pode alterar o curso. Consulte o administrador do siacua.
usernamesiacua='f874',  # Pode alterar o username.
level=3,     # level of dificulty in a scale 1-5.
slip=0.2,   # The probability of knowing how to answer, commit a mistake.
guess=0.25,  # The probability of guessing the right option without any study.
discr=0.3,   # Parameter `discr` is the probability that a student knows how to find the
concepts= [(920, 0.2),(931,0.8)], # Uma lista como [(110, 0.3),(135, 0.7)] onde 0.3+0.7
grid2x2=False,  # Write exercise answering options in a 2x2 grid (useful for graphics).
siacuatest=False )  # If True, send data to a test machine.
```



```
#FAÇA SHIFT-ENTER PARA ENVIAR PARA siacua.web.ua.pt (contacto dmat-siacua@ua.pt)

#meg.siacua(
  ekeys=[0,1,2,3,4,5,6,7,8,9],  # Cada inteiro gera um exercício diferente.
  course='matsec',  # Pode alterar o curso. Consulte o administrador do siacua.
  usernamesiacua='f874',  # Pode alterar o username.
  level=3,     # level of dificulty in a scale 1-5.
  slip=0.2,   # The probability of knowing how to answer, commit a mistake.
  guess=0.25,  # The probability of guessing the right option without any study.
  discr=0.3,   # Parameter `discr` is the probability that a student knows how to find the right answer.
  concepts= [(920, 0.2),(931,0.8)], # Uma lista como [(110, 0.3),(135, 0.7)] onde 0.3+0.7 = 1 e 110 e 135 são 3 dígitos (Assunto, Tema, Conceito).
  grid2x2=False,  # Write exercise answering options in a 2x2 grid (useful for graphics).
  siacuatest=False )  # If True, send data to a test machine.

    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    E26A36_Antidiferentiation_Parts_026_siacua
    Exercícios a consultar no SIACUA: New: 9062, New: 9063, New: 9064, New: 9065, New: 9066, New: 9067, New: 9068, New: 9069, New: 9070, New: 9071.
    Abrir http://siacua.web.ua.pt depois de entrar no curso: Gestão Professor -- Botão 'Ler Questões'
```

Figura 3.5: Exercício E26A36_Antidiferentiation_Parts_026_siacua (2)

Nas Figuras 3.6 e 3.7 encontramos a construção do exercício propriamente dita, a qual já expusemos na Secção 3.2. Na Figura 3.6 temos as escolhas múltiplas e a resolução e na Figura 3.7 temos a definição dos valores que podem ser atribuídos aos parâmetros do problema, os distratores e pequenas *packages* e comandos que são utilizados na construção das respostas (certa e erradas).

```
#FAÇA SHIFT-ENTER PARA VER O EXERCÍCIO (e inserir na base de dados quando modificado)
meg.save(r'''
%summary Primitivas; Primitivas Por Partes


Palavras-chave: Primitiva; antiderivada.

Autores: João Silva

Ano: 2017

Propósito didático: Primitivação por partes

%problem Calcular primitivas usando o método e primitivação por partes:a ln(b x)


Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"}:


<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>


%ANSWER




  Uma primitiva desta função pode ser obtida utilizando a regra de primitivação por partes.
  <p>
___

Assim,_
 \begin{eqnarray*}
  \int f1 \, dx_
  &=& a1 \, \int 1 \, \times \, \ln(a2 x)\, dx \\
  &=& a1 \, \left( x \, \ln (a2 x) - \int x \frac {a2}{a2 x} \, dx \right) \\
  &=& a1 \, \left( x \, \ln (a2 x) - x \right)\\
  &=& a1 \, x \, \left(\ln (a2 x) - 1 \right)\\
  &=&  resposta0.
 \end{eqnarray*}

 Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portanto, qualquer função do tipo
 $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
_

_
```

Figura 3.6: Exercício E26A36_Antidiferentiation_Parts_026_siacua (3)

```
_

class E26A36_Antidiferentiation_Parts_026_siacua(ExSiacua):

    def make_random(s,edict=None):
        x=var('x')

        s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado_
        s.a1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar_
        s.a2 = ur.iunif_nonset(0,9,[0,1,-1]) # parametro a multiplicar_
        s.f1 = s.a1*ln(s.a2*x) #função a integrar
        s.resposta0 =  integrate(s.f1,x)
________


        #respostas erradas
        s.re1 = diff(s.f1, x)
        s.re2 =   s.a1*x*(ln(s.a2*x - 1/(s.a2)))
        s.re3 =   s.a1/2 *(ln (s.a2*x))^2



__________
         # Controlar os Sinais
        if s.a2 >1:
           s.sgn_a2 = '-'
        else:
           s.sgn_a2 = '+'


        s.a11= abs(s.a1)
        s.a22= abs(s.a2)

__  ''')
```

Figura 3.7: Exercício E26A36_Antidiferentiation_Parts_026_siacua (4)

Após ordem de compilação (*Shift* + *Enter*), apresenta-se o problema com um uma chave em concreto que foi escolhida aleatoriamente. Neste caso, os parâmetros a_1 e a_2 foram concretizados (pela chave gerada `ekey=0`) com os valores -6 e 3, respetivamente.

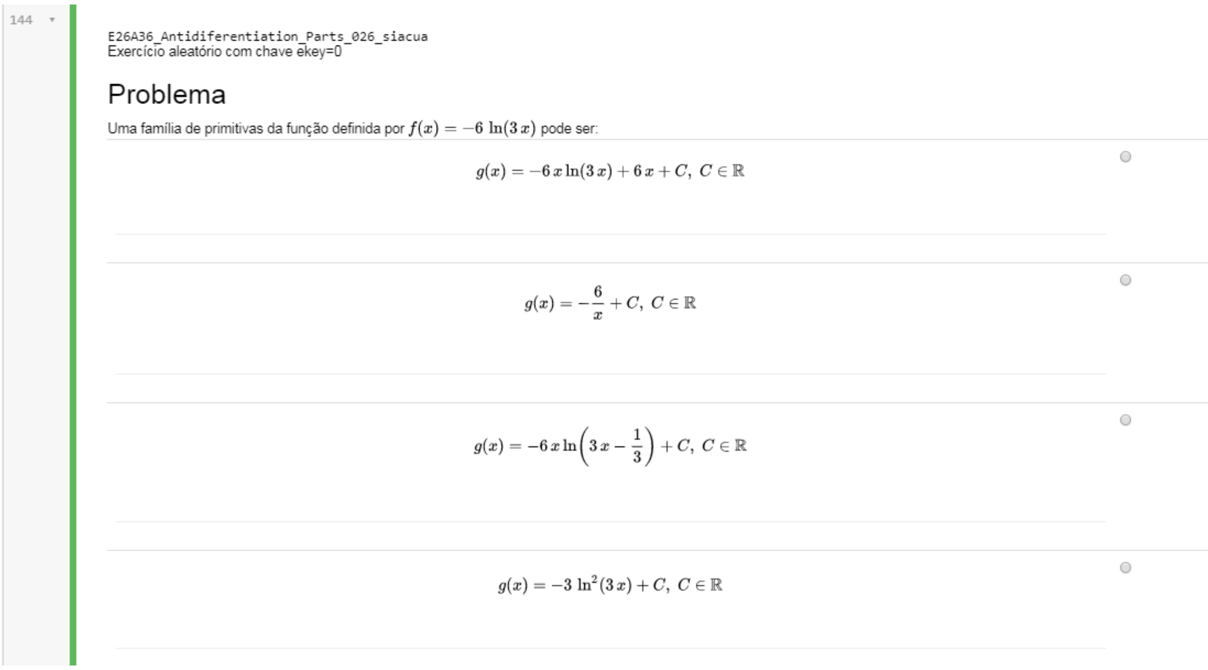


Figura 3.8: Exercício E26A36_Antidiferentiation_Parts_026_siacua (5)

A primeira resposta apresentada é a correta e as restantes são os distratores. Realçamos que esta visualização é vista apenas na ótica do programador uma vez que, para a visualização na óptica do utilizador a resposta correta é posicionada de forma aleatória conjuntamente com três distratores escolhidos também de forma aleatória.

Por fim, surge a resolução:

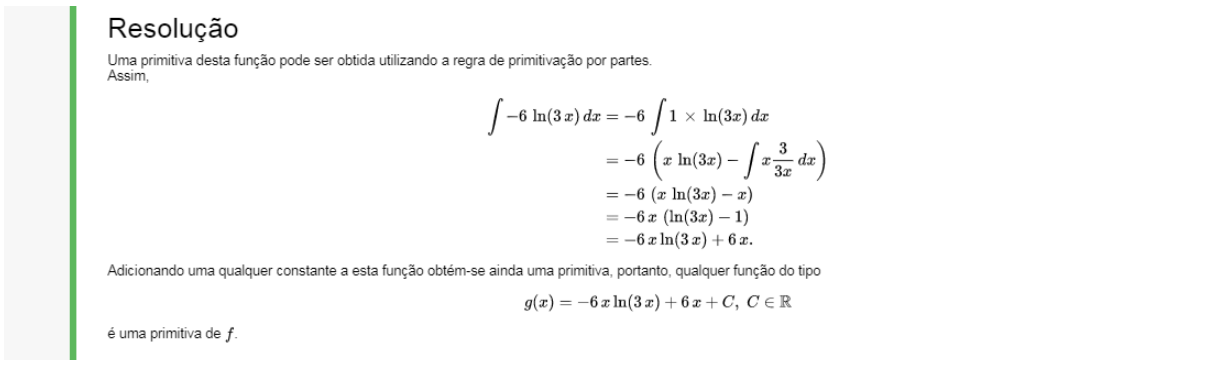


Figura 3.9: Exercício E26A36_Antidiferentiation_Parts_026_siacua (6)

3.4 Apresentação do *SIACUA*

Apresentamos a seguir uma imagem da página inicial deste sistema:

Figura 3.10: Página inicial do SIACUA

Clicando em ver cursos, obtemos a página que se apresenta na figura seguinte:

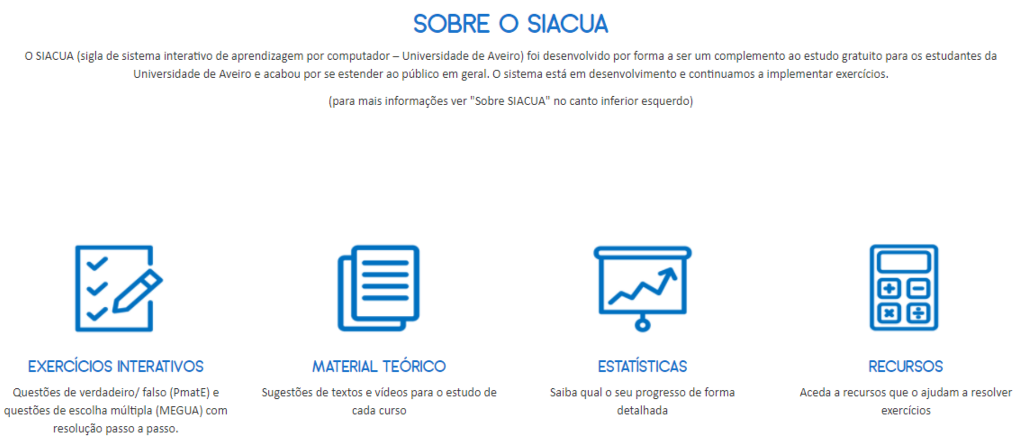


Figura 3.11: Conteúdos do SIACUA

Depois de feita a autenticação, o utlizador escolher *APRENDER* ou *PRATICAR*. No caso em apreço, deve escolher a opção *PRATICAR*.

Figura 3.12: Página de escolha entre *APRENDER* e *PRATICAR* do *SIACUA*

3.5 Apresentação do exercício no *SIACUA*

Na figura 3.13 seguinte apresentamos uma concretização de um exercício gerado pelo mesmo exercício parametrizado agora com os valores 9 e 3, para os parâmetros a_1 e a_2 respetivamente, na *webpage* onde pode ser visualizado por qualquer utilizador do *SIACUA*.

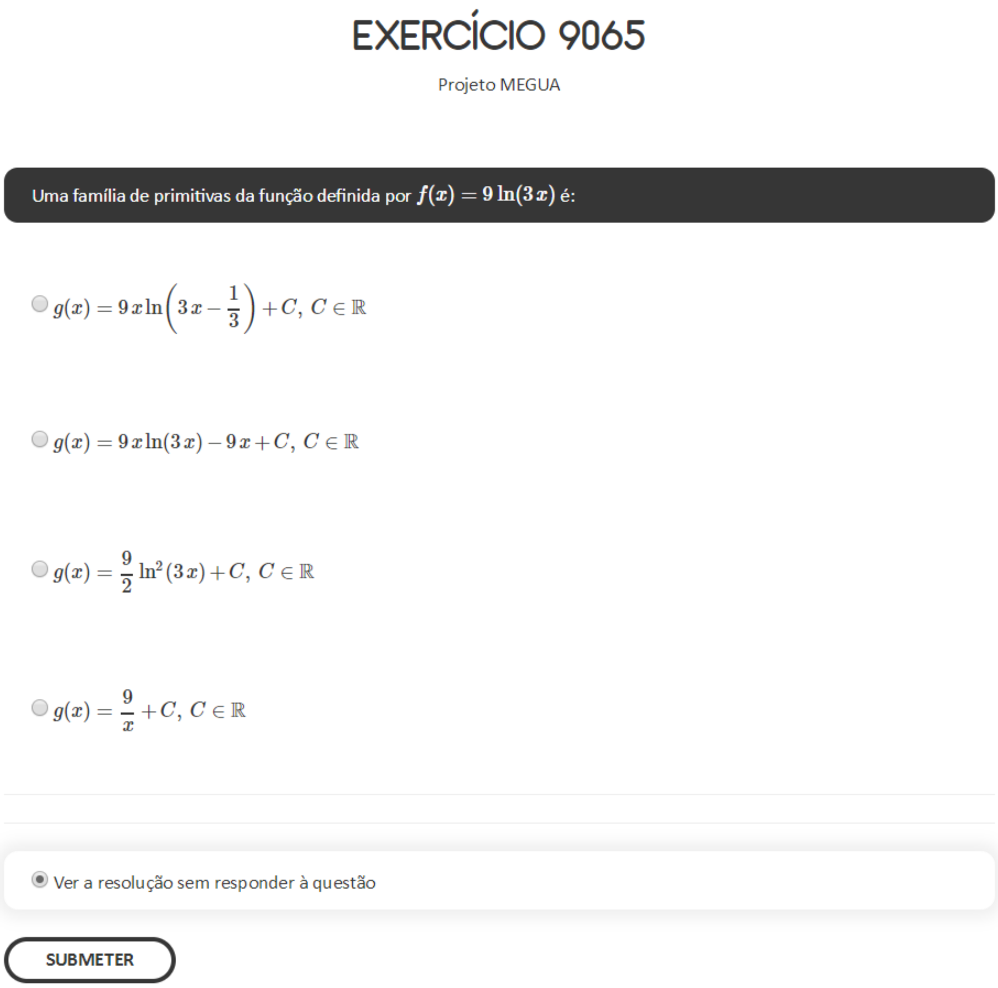


Figura 3.13: Exemplo de uma concretização de um exercício parametrizado

Na figura seguinte apresentamos a concretização anterior de um exercício parametrizado onde já foi selecionada a resposta.

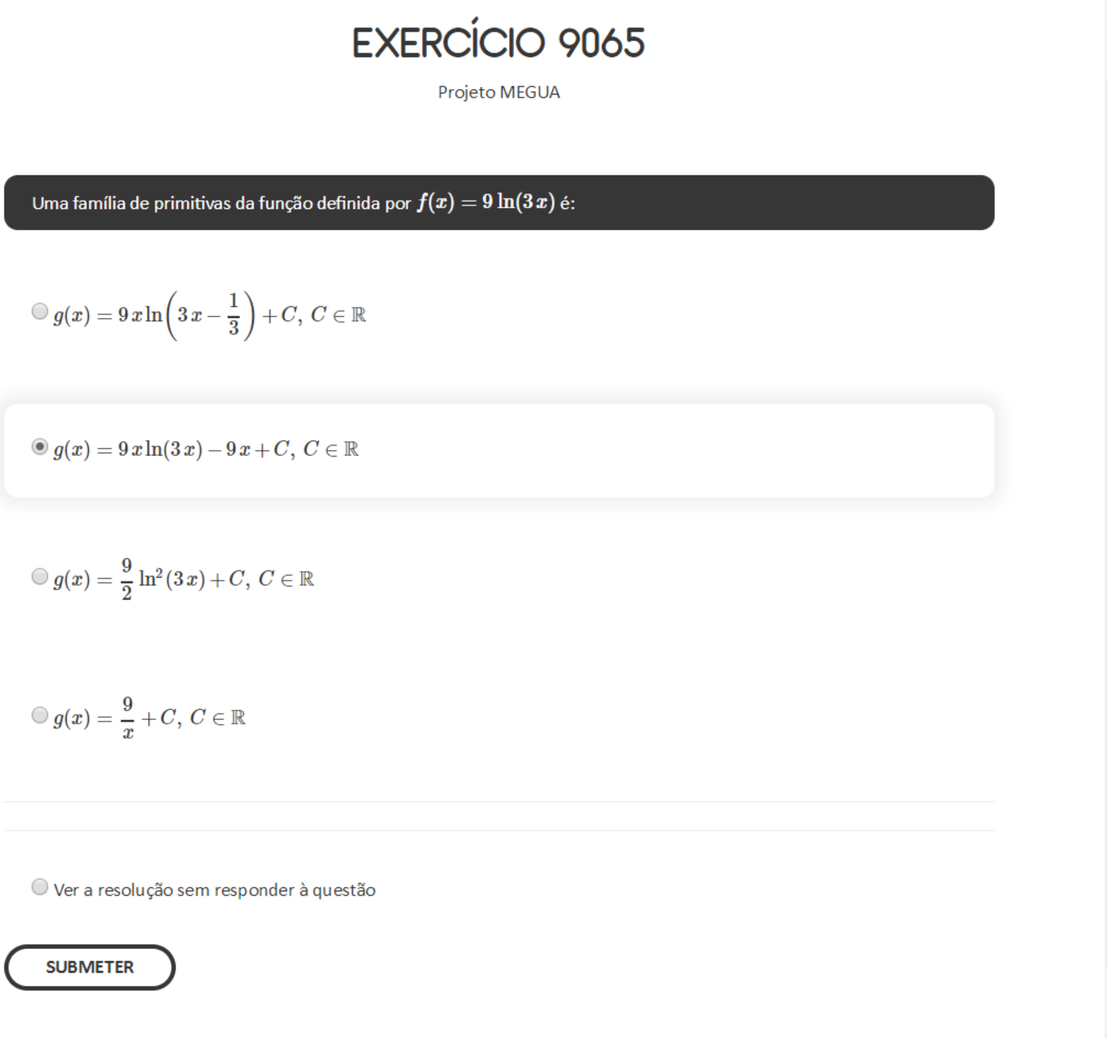


Figura 3.14: Exemplo de um exercício parametrizado com resposta selecionada

Depois de responder, o estudante visualiza a resolução apenas no caso de selecionar uma resposta errada. Se a resposta for correta, o interlocutor é informado de que pode prosseguir.

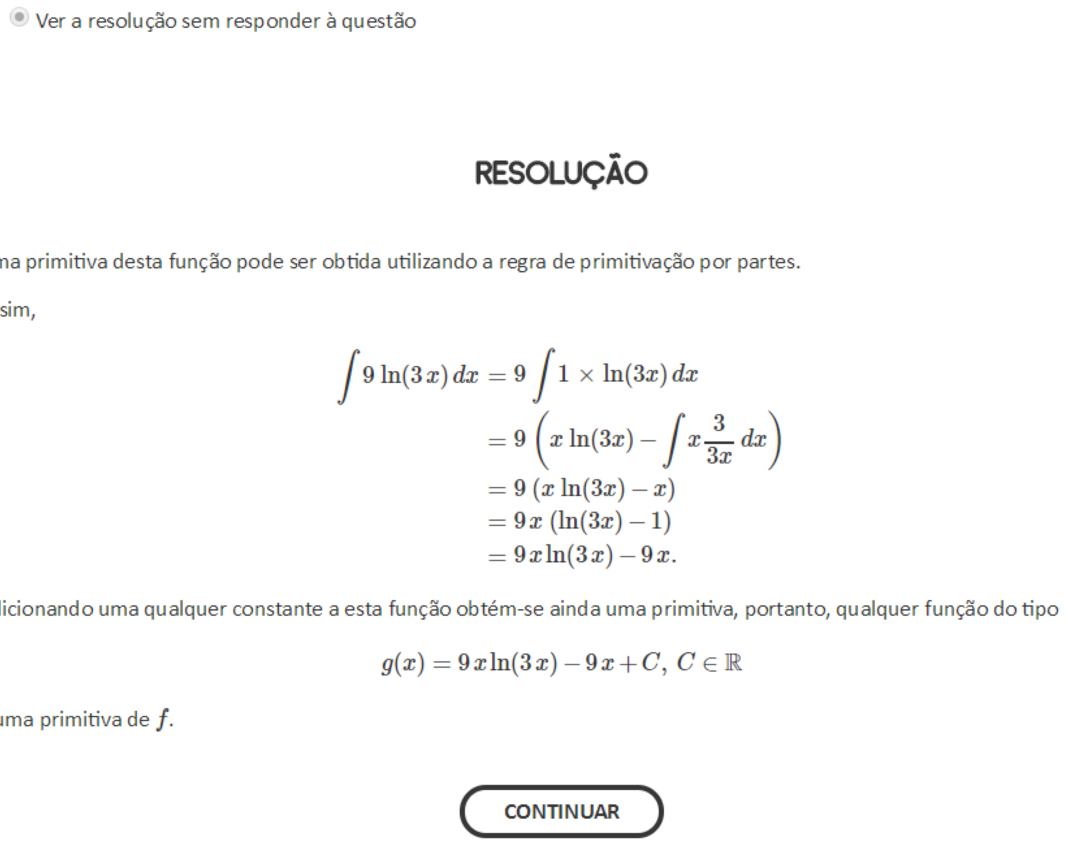


Figura 3.15: Exemplo de resolução de uma concretização de um exercício parametrizado

Conclusão

A abordagem do domínio de conteúdos *Primitivas e Cálculo Integral* encontra terreno fértil e faz sentido numa altura em que esta unidade temática passou a fazer parte do novo *Programa Curricular de Matemática A* para o Ensino Secundário. No entanto, e apesar da importância que lhe foi atribuída, o seu ensino poderá estar comprometido. De facto, nas *Orientações de gestão curricular para o Programa e Metas Curriculares de Matemática A*, está previsto que o mesmo possa ser facultativo nos anos letivos de 2017/2018 e 2018/2019, dadas as dificuldades e atrasos inerentes à implementação de um novo Programa [9]. Acresce ainda o facto de, neste período de transição entre Programas, existirem alunos do antigo Programa a realizar o Exame Nacional de Matemática A. Como o referido domínio não fez parte do seu currículo, o mesmo não será avaliado nesta prova, razão pela qual muitos dos professores que lecionam o 12.º ano de escolaridade terem optado por não incluí-lo nas suas planificações anuais. Apesar desta tendência, não podemos deixar de notar que é chamada a atenção dos professores para a importância de trabalhar o domínio em causa também nos anos letivos de exceção, sempre que seja possível e ainda que o tempo apenas permita fazê-lo parcialmente [9]. Mais uma vez se reforça a ideia de que o estudo dos conteúdos deste domínio contribui para a consolidação de outros conteúdos centrais do Ensino Secundário e o facto de uma primeira abordagem da noção de primitiva e de integral ao nível do Ensino Secundário poder ser uma mais-valia para um prosseguimento de estudos de sucesso no Ensino Superior [9, 10].

Este trabalho convergiu para este fim e abre portas a uma abordagem futura por parte de alunos e professores, tentando desmistificar a dificuldade que se associa ao domínio de conteúdos *Primitivas e Cálculo Integral*. Sendo realizado no hiato que antecede a obrigatoriedade do seu ensino no Ensino Secundário, a sua utilização será inicialmente direcionada a alunos do Ensino Superior, esperando-se que a mesma se estenda, tão breve quanto possível, aos alunos e professores do Ensino Secundário, o que o torna transversal aos dois níveis de ensino. A criação de recursos digitais visou sobretudo o estudo autónomo por parte dos alunos do domínio *Primitivas e Cálculo Integral*, o que não exclui que os professores utilizem os mesmos recursos em contexto de sala de aula. Assim, cabe ao professor, acima de tudo orientar o estudo, cabendo depois ao aluno, em interação com o computador, a construção do próprio conhecimento. O ritmo e a direção com que o conhecimento é construído são definidos por cada estudante. Neste sentido, um sistema como o *SIACUA* permite a cada aluno um percurso de construção do conhecimento adaptado a si e às suas próprias idiossincrasias. Partindo do pressuposto que as tecnologias da informação e da comunicação alteraram e continuam a alterar estratégias e métodos pedagógico-didáticos, reconhecemos que as mesmas se constituem simultaneamente um facilitador e um acelerador das aprendizagens. Por outro lado, o recurso por parte dos alunos aos exercícios parametrizados que construímos permite-lhes resolver exercícios idênticos ou distintos em diferentes etapas do seu estudo, sempre com acesso à respetiva resolução detalhada, e estando acessíveis em qualquer dispositivo com ligação à internet. Consideramos pois que o material assim desenvolvido se torna motivador e acessível a todos quanto o procurem. Convem ainda referir que a base de dados de exercícios não se considera terminada na medida em que, dada a interatividade que a caracteriza, muitos outros exercícios podem ser criados, modificados ou melhorados em função da necessidade de diferentes abordagens. Desta forma, este trabalho não se esgota aqui, podendo ser um ponto de partida para outros estudos.

A realização deste trabalho foi um desafio constante e implicou muita dedicação, muito empenho e muito esforço. O facto de termos de aprender de raiz várias linguagens de programação utilizadas em diferentes fases do trabalho, uma vez que a formação pessoal nestas áreas era inexistente, obstaculizou por diversas vezes um trabalho que se desejava mais fluído e contínuo. A construção de exercícios revelou-se uma tarefa árdua dado que cada novo tipo de exercícios tinha como premissa um conjunto de diferentes procedimentos do código a utilizar. Mas, pese embora todas as dificuldades, este trabalho promoveu muita aprendizagem, nomeadamente na programação em LaTeX. Para além, disso fomentou o contacto e posterior domínio de novas realidades informáticas e tecnológicas que não eram por nós conhecidas e permitiu ainda perceber até que ponto um sistema como o *SIACUA* pode catalisar a aprendizagem, tanto em rapidez como em direção: o estudante pode otimizar a aprendizagem uma vez que estuda e pratica exatamente o que necessita, quando necessita, ao invés de um estudo tradicional em que o estudante "consumia" todo um "pacote" de informação em que alguma, ou muita da qual, era desinteressante para o momento.

Posto isto, consideramos que esta foi uma experiência muito motivadora e sobretudo enriquecedora. Pensamos que a tarefa de construção de exercícios com as tecnologias informáticas que tivemos ao dispor contribuiu para que as *performances* de aprendizagem de quem as utiliza sejam significativamente melhoradas, seja em quantidade, em qualidade ou em velocidade, o que se traduz no cumprimento do objetivo e do propósito deste trabalho, contribuindo para um melhor ensino da Matemática, em concreto do domínio de conteúdos *Primitivas e Cálculo Integral*.

Bibliografia

[1] C. Andrade, P. Pereira, P. Pimenta *Novo Ípsilon 12, Volume 3, Matemática A 12.º ano, ensino Secundário*, Raiz Editora, 2017.

[2] A. Bento. *Apontamentos da disciplina de Cálculo I.* Universidade da Beira Interior, Ano Lectivo 2011/2012.

[3] B. Costa, E. Rodrigues *Novo Espaço Parte 2, Matemática A 12.º ano*, Porto Editora, 2017.

[4] J. Ferreira *Introdução à Análise Matemática* 2.ª edição 1988

[5] T. Maciel *A História da Matemática e o estabelecimento de elos entre o Ensino Superior e a Educação Básica*, Universidade Federal da Paraíba, `http://rei.biblioteca.ufpb.br/jspui/bitstream/123456789/369/1/TSM08072013.pdf`, 2011.

[6] *Mathematics Exercise Generator, Universidade de Aveiro (Megua)*, disponível em `http://cms.ua.pt/megua/`.

[7] M. Neves, L. Gameiro, A. Silva *Máximo Complexos e Primitivas, Matemática A 12.º ano*, Porto Editora, 2017.

[8] M. Oliveira. *Apontamentos da disciplina de Análise Matemática I.* Instituto Politécnico de Tomar, Ano Lectivo 2007/2008.

[9] *Orientações de gestão curricular para o Programa e Metas Curriculares de Matemática A*, `http://www.dge.mec.pt/sites/default/files/Secundario/Metas`, Ministério da Educação, 2016.

[10] *Programa e Metas Curriculares de Matemática A*, disponível em `http://www.dge.mec.pt/sites/default/files/Secundario/Documentos`, Ministério da Educação, 2013.

[11] M. Rosa. *Apontamentos de Análise Matemática*. Instituto Politécnico da Guarda, 2014.

[12] C. Sarrico. *Cálculo Diferencial e Integral para Funções de Várias Variáveis*. Esfera do Caos, 2009.

[13] *Siacua (Interactive Computer Learnig System, University of Aveiro)*, disponível em `http://siacua.web.ua.pt`.

[14] W. A. Stein et al. *Sage Mathematics Software (Version 4.6.1), The Sage Development Team.*, `http://www.cocalc.org`, 2011.

[15] J. Stewart. *Calculus - Early transcendentals*. 8th Edition, 2015.

[16] P. Urban, J. Owen, D. Martin, R. Haese, S. Haese, M. Bruce *Mathematics for the international student, International Baccalaureate Mathematics HLCourse*, Haese & Harris Publications, 2006.

Apêndice A

Exercícios produzidos

Neste anexo apresentam-se todos os exercícios criados, com as várias componentes, incluindo o texto da questão, as várias opções, a resolução e o código em *Python*. Os exercícios 1 a 7 já haviam sido criados e foram anexados a este trabalho para evitar a duplicação dos mesmos. Os restantes 19 exercícios foram criados por nós durante a elaboração desta monografia.

E26A36_Antidiferentiation_Imediata_001_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas


Palavras-chave: Primitiva; antiderivada; potência.
Autores: Paula Carvalho
Ano: 2017
Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas: potência de expoente inteiro (excepto -1,0,1)


Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"


<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
```

```
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>

%ANSWER

<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula e

<showone res0>
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um númer
obtém-se $$g'(x) = f1, $$ portanto $g(x) = resposta0  + C $ é uma família de primitivas
</thisone>
<thisone Caso 1 - (isto é comentário)>
Uma primitiva desta função pode ser obtida usando a regra de primitivação de uma potênci
$$\int \, x^n \, dx=\frac{x^{n+1}}{n+1}.$$
Assim,
\begin{eqnarray*}
\int \left(f1 \right)\, dx &=& a1 \int \, \left(x^{exp1} \right)\, dx\\
&=& a1 \frac{x^{exp1+1}}{exp1+1}\\
&=&  resposta0.
\end{eqnarray*}
Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portanto,
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
</thisone>
</showone>

class E26A36_Antidiferentiation_Imediata_001_siacua(ExSiacua):
```

```
def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressao a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1
s.exp1= ur.iunif_nonset(-9,9,[0,1,-1]) #expoente inteiro
#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opcao lista
s.res0   = ur.iunif(0,1) #opcao de escrita da resolução 0,1  (2 possibilidades)

s.f1=s.a1*x^s.exp1
s.resposta0 = integrate( s.f1, x)

#respostas
s.re1 = diff(s.f1, x)          # derivada
s.re2 = s.exp1*s.f1            # produto do expoente pela funçao
s.re3 = s.resposta0 *(s.exp1+1) # esquece a divisao por n+1

''')
```

E26A36_Antidiferentiation_Imediata_002_siacua

```
meg.save(r'''
%SUMMARY  Primitivas; Primitivas quase imediatas; Função potência



Palavras chave: Primitiva, antiderivada, potência; radical


Autores: Paula Oliveira -  Tópicos de Matemática - CTESP e Matemática 1;
Paula Carvalho(Renomeado. Era E26A36_Antidiferentiation_Imediata_002_siacua.sagews)


Ano: 2016; 2017
```

```
Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)

%PROBLEM Primitivas de potência racional $u^n$ com u=ax+b e n=1/2

Indique a família de primitivas de
\[
f(x)=\sqrt{onf}.
\]

<multiplechoice>

<choice>
<p> \[
\int{\sqrt{onf} \, dx}=onfip + K, \mbox{ com } K \in \mathbb{R}
\]</p>

</choice>

<choice>
<p> \[
\int{\sqrt{onf} \, dx}=w1 + K, \mbox{ com } K \in \mathbb{R}
\]</p>


</choice>
```

```
<choice>
<p> \[
\int{\sqrt{onf} \, dx}=w2 + K, \mbox{ com } K \in \mathbb{R}
\]</p>
</choice>

<choice>
<p> \[
\int{\sqrt{onf} \, dx}=w3 + K, \mbox{ com } K \in \mathbb{R}
\]</p>
</choice>

</multiplechoice>

%ANSWER

Observe-se que a expressão de $f(x)$ pode ser escrita como potência
\[
f(x)=\left(onf\right)^{\frac{1}{2}}
\]
podendo aplicar-se a regra da potência, $\displaystyle  \int\,(u^n\, u’)\, dx=\frac{u
\[
\int{\sqrt{onf} \, dx}=\int{\left(onf\right)^{\frac{1}{2}} \, dx}=ona\int{ina\left(on
ona \, \frac{\left(onf\right)^{\frac{1}{2}+1}}{\frac{1}{2}+1}+K
\]
Assim,
\[
\int{\sqrt{onf} \, dx}=onfip + K, \mbox{ com } K \in \mathbb{R}.
\]
```

```
class E26A36_Antidiferentiation_Imediata_002_siacua(ExSiacua):
"""este exercicio calcula a primitiva de sqrt(ax+b) """
def make_random(self):
x=var('x')
y=var('y')
self.ina=ur.iunif_nonset(-10,10,[0,1,-1])
self.inb=ur.iunif_nonset(-10,10,[0])




self.ona = 1/self.ina

#definir a função base sqrt(ax+b)
self.onf=self.ina*x+self.inb
self.onfi=sqrt(self.onf)
self.onfil=latex(self.onfi)
self.onfip=integral(self.onfi,x)
self.onfipl=latex(self.onfip)
#wrong answers
self.w1=self.onfip*self.ina
self.w2=self.ina/self.onfi
self.w3=(self.ina/2)/self.onfi

''')
```

E26A36_Antidiferentiation_Imediata_003_siacua

```
meg.save(r'''
%SUMMARY  Primitivas; Primitivas quase imediatas; Função potência


Palavras chave: Primitiva; antiderivada; potência; radical
```

```
Autores: Paula Oliveira -  Tópicos de Matemática - CTESP e Matemática 1;
Paula Carvalho(Renomeado. Era E26A36_antiderivative_003_siacua.sagews)

Ano: 2016; 2017

Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%PROBLEM Primitiva potência com radical (expoente -1/2)

A família de primitivas da função
\[
g(x)=ongil
\]

é:

<multiplechoice>

<choice>
<p> \[
\int{\frac{1}{\sqrt{onf}} \, dx}=ongip+K, \mbox{ com } K \in \mathbb{R}
\]</p>

</choice>

<choice>
<p> \[
\int{\sqrt{onf} \, dx}=w1 + K, \mbox{ com } K \in \mathbb{R}
\]</p>
```

```
</choice>

<choice>
<p> \[
\int{\sqrt{onf} \, dx}=w2 + K, \mbox{ com } K \in \mathbb{R}
\]</p>
</choice>

<choice>
<p> \[
\int{\sqrt{onf} \, dx}=w3 + K, \mbox{ com } K \in \mathbb{R}
\]</p>
</choice>
</multiplechoice>

%ANSWER

Como a função $g$ se pode escrever como potência, aplica-se a regra de primitivação da p
\begin{eqnarray*}
\int{\frac{1}{\sqrt{onf}} \, dx}
&= & \int{\left(onf\right)^{-\frac{1}{2}} \, dx}\\
&= & ona \int{ina\left(onf\right)^{-\frac{1}{2}} \, dx}\\
&= & ona \, \frac{\left(onf\right)^{-\frac{1}{2}+1}}{-\frac{1}{2}+1}+K\\
&= & ona \, \frac{\left(onf\right)^{\frac{1}{2}}}{\frac{1}{2}}+K\\
\end{eqnarray*}
Portanto,
\[
\int{\frac{1}{\sqrt{onf}} \, dx}=ongip+K, \mbox{ com } K \in \mathbb{R}.
\]
```

```
class E26A36_Antidiferentiation_Imediata_003_siacua(ExSiacua):
"""este exercicio calcula a primitiva de  1/sqrt(ax+b)"""
def make_random(self):
x=var('x')
y=var('y')
self.ina=ur.iunif_nonset(-10,10,[0,1,-1])
self.inb=ur.iunif_nonset(-10,10,[0])




self.ona = 1/self.ina
self.onb = self.inb
#definir a função base sqrt(ax+b)
self.onf=self.ina*x+self.inb
#definir a função base 1/sqrt(ax+b)
self.ongi=1/sqrt(self.onf)
self.ongil=latex(self.ongi)
self.ongip=integral(self.ongi,x)
self.ongipl=latex(self.ongip)
#wrong answers
self.w1=self.ongip*self.ina
self.w2=self.ina*(-3/2)/(sqrt(self.onf^3))
self.w3=self.ona*sqrt(self.onf)


''')
```

E26A36_Antidiferentiation_Imediata_004_siacua

```
meg.save(r'''
%SUMMARY  Primitivas; Primitivas quase imediatas; Função potência



Palavras chave: Primitiva, antiderivada, potência





Autores: Paula Oliveira -  Tópicos de Matemática - CTESP e Matemática 1;
Paula Carvalho(Renomeado. Era E26A36_antiderivative_001_siacua.sagews)

Ano: 2016; 2017

Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)

%PROBLEM Primitiva de potência $ 1/ (ax+b)^n $

Indique a família de primitivas da função
<showone exp1>
<thisone>
\[
f(x)=(onf)^{ini}
\]
</thisone>
<thisone>
\[
f(x)=\frac{1}{(onf)^{ini1}}
\]
</thisone>
</showone>
```

```
<multiplechoice>

<choice>
<showone correct1>
<thisone>
<p> \[
\int{\left(onf\right)^{ini} \, dx}=onai \left(onf\right)^{oni}+K, \mbox{ com } K \in \
\]</p>
</thisone>
<thisone>
<p> \[
\int{\frac{1}{(onf)^{ini1}} \, dx}=onai \frac{1}{\left(onf\right)^{oni1}}+K, \mbox{ co
\]</p>
</thisone>
</showone>
</choice>

<choice><showone wrong1>
<thisone>
<p> \[
\int{\left(onf\right)^{ini} \, dx}=w11 \left(onf\right)^{w11}+K, \mbox{ com } K \in \m
\]</p>
</thisone>
<thisone>
<p> \[
\int{\frac{1}{(onf)^{ini1}} \, dx}= \frac{w11}{\left(onf\right)^{w12}}+K, \mbox{ com }
\]</p>
</thisone>
</showone>
</choice>
```

```
<choice><showone wrong2>
<thisone>
<p> \[
\int{\left(onf\right)^{ini} \, dx}=oni12 \left(onf\right)^{oni}+K, \mbox{ com } K \in \m
\]</p>
</thisone>
<thisone>
<p> \[
\int{\frac{1}{(onf)^{ini1}} \, dx}= \frac{oni12}{\left(onf\right)^{oni1}}+K, \mbox{ com
\]</p>
</thisone>
</showone></choice>

<choice><showone wrong3>
<thisone>
<p> \[
\int{\left(onf\right)^{ini} \, dx}=c1 \left(onf\right)^{w11}+K, \mbox{ com } K \in \math
\]</p>
</thisone>
<thisone>
<p> \[
\int{\frac{1}{(onf)^{ini1}} \, dx}= \frac{c1}{\left(onf\right)^{w12}}+K, \mbox{ com } K
\]</p>
</thisone>
</showone></choice>

</multiplechoice>

%ANSWER
```

```
Aplicando a regra da potência, $\displaystyle \int (u^n\, u') dx
= \frac{u^{n+1}}{n+1}$, com $u=onf$ e $n=ini$, vem
\begin{eqnarray*}
\int{\left(onf\right)^{ini} \, dx}
&=& ona \int{ina\left(onf\right)^{ini} \, dx}\\
&=& ona \frac{\left(onf\right)^{ini+1}}{ini+1}+K \\
\end{eqnarray*}
Então,
\[
\int{\left(onf\right)^{ini} \, dx}=onai \left(onf\right)^{oni}+K, \mbox{ com } K \in \
\]
```

```
class E26A36_Antidiferentiation_Imediata_004_siacua(ExSiacua):
"""este exercicio a primitiva de uma potência inteira em que a base é ax+b"""
def make_random(self):
x=var('x')
y=var('y')
self.ina=ur.iunif_nonset(-10,10,[0,1])
self.inb=ur.iunif_nonset(-10,10,[0])
self.ini=ur.iunif_nonset(-10,10,[0,1,-1,2])
self.ini1=-self.ini
if self.ini>0:
self.exp1=0
self.correct1=0
self.wrong1=0
self.wrong2=0
self.wrong3=0
else:
self.exp1=1
self.correct1=1
```

```
self.wrong1=1
self.wrong2=1
self.wrong3=1

self.ona = 1/self.ina
self.oni = self.ini+1
self.oni1=-self.oni
self.oni12=1/self.oni
self.onai=self.ona/self.oni
#definir a função base
self.onf=self.ina*x+self.inb
self.onfi=self.onf^self.ini
self.onfip=integral(self.onfi,x)
self.onfipf=factor(self.onfip)
self.onfipl=latex(self.onfipf)
self.c1=self.oni*self.ina
#wrong answers
#WRONG 1
self.w11=self.ini-1
self.w12=-self.w11
self.w1=self.w11*(self.onf)^self.w11
#wrong 2
self.w2=(self.onf)^(self.ini+1)/(self.ini+1)
#wrong 3
self.w3=self.ini*(self.onf)^(self.w11)*self.ina

''')
```

E26A36_Antidiferentiation_Parts_005_siacua

```
meg.save(r'''
%SUMMARY  Primitivas; Primitivas por partes
```

Palavras chave: Primitiva, antiderivada, por partes, potência

Autores: Paula Oliveira Matemática 1 - TSP (adaptado);
Paula Carvalho(Renomeado. Era E26A36_antiderivativeparts_005_siacua.sagews)

Ano: 2016; 2017

Propósito didático: Primitivação por partes

```
%PROBLEM Potencia Primitiva de potência $ (cx)(ax+b)^n $

<h3>Indique a família de primitivas de</h3>
\[
\int{ongl\,(onfl)^{ini} \, dx}
\]



<multiplechoice>

<choice><p> \[
\int{ongl\,(onfl)^{ini} \, dx}= onb3l \,x (onfl)^{oni1} -onb4l \, (onfl)^{oni2}+C, \;
\] </p></choice>

<choice><p> \[
\int{ongl\,(onfl)^{ini} \, dx}=w1+C, \; C \in \mathbb{R}
\] </p></choice>

<choice><p>\[
\int{ongl\,(onfl)^{ini} \, dx}=w2+C, \; C \in \mathbb{R}
```

```
\]</p></choice>

<choice><p> \[
\int{ongl\,(onfl)^{ini} \, dx}=w3+C, \; C \in \mathbb{R}
\] </p></choice>

</multiplechoice>

%ANSWER

Usando o método de primitivação por partes
\[
\int{f'(x) \, g(x) \, dx}=f(x)\, g(x)- \int{f(x) \, g'(x)\,dx}
\]
considera-se
\[
\begin{array}{lll}
f'(x)=(onfl)^{ini} & \quad & \displaystyle f(x)=onb1l \, \frac{(onfl)^{oni1}}{oni1}=onb2
& & \\
g(x)=ongl & \quad &\displaystyle  g'(x)=ongdl
\end{array}.
\]
Então,
\[
\begin{array}{lll}
\displaystyle \int{ongl\,(onfl)^{ini} \, dx}&=&\displaystyle ongl \, onb2l \, (onfl)^{on
&  & \\
&=&\displaystyle onb3l \,x (onfl)^{oni1} -onb3l \, \frac{(onfl)^{oni2}}{oni2}+C\\
& & \\
&=&\displaystyle onb3l \,x (onfl)^{oni1} -onb4l \, (onfl)^{oni2}+C, \, C \in \mathbb{R}
\end{array}
```

```
\]


class E26A36_Antidiferentiation_Parts_005_siacua(ExSiacua):
"""este exercicio usa a primitivação por partes para calcular a primitiva de (ax)(bx+c
def make_random(self):
x=var('x')
y=var('y')
self.ina=ur.iunif_nonset(2,10,[0,1])
self.inb=ur.iunif_nonset(2,10,[0])
self.inc=ur.iunif_nonset(2,10,[0,1,-1])
self.ini=ur.iunif(3,9)

self.ona = self.ina
self.onb = self.inb
self.onc = self.inc
#definir a função f
self.onf=self.inb*x+self.inc
self.onfl=latex(self.onf)
#definir a primitiva de f
self.oni1=self.ini+1
self.oni2=self.ini+2
self.onb1=1/self.inb
self.onb1l=latex(self.onb1)
self.onb2=self.onb1/self.oni1
self.onb2l=latex( self.onb2)
#definir a função g
self.ong=self.ina*x
self.ongl=latex(self.ong)
self.ongd=derivative(self.ong,x)
self.ongdl=latex(self.ongd)
```

```
self.onb3=self.onb2*self.ongd
self.onb3l=latex(self.onb3)
self.onb4=self.onb3/self.oni2
self.onb4l=latex(self.onb4)

#wrong answers
self.auxw1=integral(self.ong,x)
self.auxw2=integral(self.onf^self.ini,x)
self.w1=self.auxw1*self.auxw2
self.w2=self.ong*self.onf^(self.ini+1)/(self.ini+1)-self.ina*self.onf^(self.ini+2)/(self
self.w3=self.ina*self.onf^self.ini+self.ina*x*self.ini*self.inb*self.onf^(self.ini-1)

''')
```

E26A36_Antidiferentiation_Imediata_006_siacua

```
meg.save(r'''
%SUMMARY  Primitivas; Primitivas quase imediatas; Função potência



Palavras chave: Primitiva; antiderivada; potência;


Autores: Paula Oliveira -  Tópicos de Matemática - CTESP e Matemática 1;
Paula Carvalho(Renomeado. EraE26A36_antiderivative_004_siacua.sagews)


Ano: 2016; 2017


Propósito didático: Primitivação por imediata




%PROBLEM Potência da função seno $sin^n(ax) cos (ax)$
```

```
Indique a opção que traduz a família de primitivas de
\[
f(x)=onfi .
\]


<multiplechoice>

<choice>
<p> \[
\int{onfi \, dx}=onfip+K, \; \mbox{ com }\; K \in \mathbb{R}.
\]
</p>

</choice>

<choice>
<p> \[
\int{onfi \, dx}=w1 + K, \mbox{ com } K \in \mathbb{R}
\]</p>


</choice>

<choice>
<p> \[
\int{onfi\, dx}=w2 + K, \mbox{ com } K \in \mathbb{R}
\]</p>
</choice>

<choice>
```

```
<p> \[
\int{onfi\, dx}=w3 + K, \mbox{ com } K \in \mathbb{R}
\]</p>
</choice>
</multiplechoice>

%ANSWER

Para calcular esta primitiva usa-se a regra da potência, $\displaystyle  \int \left(u^n

<p>

Como $u'=onfi1d$, tem-se
\[
\int{(onfi1)^{ini}\, onfi2 \, dx}=ona1 \int{ina (onfi1)^{ini}\, onfi2 \, dx}=ona1 \frac{
\]
com $K \in \mathbb{R}$.


class E26A36_Antidiferentiation_Imediata_006_siacua(ExSiacua):
"""este exercicio a primitiva de uma potência inteira em que a base é sin(ax)"""
def make_random(self):
x=var('x')
y=var('y')
self.ina=ur.iunif_nonset(-10,10,[0,1,-1])
self.inb=ur.iunif_nonset(-10,10,[0,1,-1])
self.ini=ur.iunif_nonset(-10,10,[0,1,-1])


self.ona1 = 1/self.ina
self.ini1=self.ini+1
```

```
self.onr1=self.ona1/self.ini1
self.onb = self.inb
#definir a função base
self.onfi1=sin(self.ina*x)
self.onfi1d=derivative(self.onfi1)
self.onfi2=cos(self.ina*x)
self.onfi=(sin(self.ina*x))^self.ini*cos(self.ina*x)
self.onfil=latex(self.onfi)
self.onfip=integral(self.onfi,x)
self.onfipf=factor(self.onfip)
self.onfipl=latex(self.onfipf)
#wrong answers
self.w1=sin(self.ina*x)^(self.ini1)/self.ini1
self.w2=(self.ini-1)*self.ina*sin(self.ina*x)^(self.ini-1)
self.w3=self.ona1*sin(self.ina*x)^(self.ini1)

#função para reescrever   potências de funções trigonométricas e log-> ln
def rewrite(self,text):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\((([\d\\, x]+?)\\right\)\^\{(\d+)\}'  ,   r'\sin^{\2}(\1)'  ),
( ur'\\cos\\left\((([\d\\, x]+?)\\right\)\^\{(\d+)\}'  ,   r'\cos^{\2}(\1)'  ),
( ur'\\log'  ,   r'\\ln'  )
]

#Esta parte é sempre a mesma.
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text
```

```
''')
```

E26A36_Antidiferentiation_Parts_007_siacua

```
meg.save(r'''
%SUMMARY  Primitivas; Primitivas por partes



Palavras chave: Primitiva; antiderivada; potência;


Autores: Paula Oliveira -  Matemática 1 - TSP;
Paula Carvalho(Renomeado. Era E26A36_antiderivativeparts_003_TSP_siacua.sagews)


Ano: 2016; 2017


Propósito didático: Primitivação por partes: $(ax+b) cos (cx)$



%PROBLEM Polinómio e cosseno  $(ax+b) cos (cx)$


Determine a família de primitivas
\[
\int{(ongl) \, onfdl \, dx}
\]




<multiplechoice>


<choice><p> \[
\int{(ongl) \, onfdl \, dx}=onhfacl+C, \; C \in \mathbb{R}
\] </p></choice>
```

```
<choice><p> \[
\int{(ongl) \, onfdl \, dx}=w1+C, \; C \in \mathbb{R}
\] </p></choice>

<choice><p> \[
\int{(ongl) \, onfdl \, dx}=w2+C, \; C \in \mathbb{R}
\]</p></choice>

<choice><p> \[
\int{(ongl) \, onfdl \, dx}=w3+C, \; C \in \mathbb{R}
\] </p></choice>

</multiplechoice>

%ANSWER

Usando o método de primitivação por partes
\[
\int{f'(x) \, g(x) \, dx}=f(x)\, g(x)- \int{f(x) \, g'(x)\,dx}
\]
considera-se
\[
\begin{array}{lll}
f'(x)=onfdl & \quad & \displaystyle f(x)=onfl\\
g(x)=ongl & \quad & g'(x)=ongdl
\end{array}.
\]
Assim,
\[
\begin{array}{lll}
```

```
\displaystyle \int{(ongl) \, onfdl \, dx}&=&
\displaystyle onfl \, (ongl)-\int{  onfgdl  \, dx}\\
& & \\
&=&
\displaystyle onfl \, (ongl)-\left(onfgdil\right)+C
\end{array}
\]
Portanto,
\[
\int{(ongl) \, onfdl \, dx}=onhfacl+C, \; C \in \mathbb{R}
\]
```

```
class E26A36_Antidiferentiation_Parts_007_siacua(ExSiacua):
"""este exercicio usa a primitivação por partes para calcular a primitiva de (ax+b)cos(c
def make_random(self):
x=var('x')
y=var('y')
self.ina=ur.iunif_nonset(-10,10,[0,1])
self.inb=ur.iunif_nonset(-10,10,[0])
self.inc=ur.iunif_nonset(2,10,[0,1,-1])

self.ona =self.ina
self.onb = self.inb
self.onc = self.inc
#definir a função um
self.onfd=cos(self.inc*x)

#função fd em latex
self.onfdl=latex(self.onfd)
```

```
self.onf=integral(self.onfd,x)
#função f em latex
self.onfl=latex(self.onf)

#definir a função dois
self.ong=self.ina*x+self.inb
self.ongd=derivative(self.ong,x)
#funcoes g e gd em latex
self.ongl=latex(self.ong)
self.ongdl=latex(self.ongd)
#função integranda partes
self.onfgd=self.onf*self.ongd
#função integranda partes latex
self.onfgdl=latex(self.onfgd)
self.onfgdi=integral(self.onfgd,x)
#função integranda partes latex
self.onfgdil=latex(self.onfgdi)
#Função integranda
self.onfgi=self.onfd*self.ong
#Função integranda latex
self.onfgil=latex(self.onfgi)
#Função final h
self.onh=integral(self.onfgi,x)
self.onhl=latex(self.onh)
self.onhfac=factor(self.onh)
self.onhfacl=latex(self.onhfac)
#wrong answers
self.auxw1=integral(self.ong,x)
self.auxw2=integral(self.onfd,x)
self.w1=self.auxw1*self.auxw2
self.w2=self.ong*sin(self.inc*x)+self.ina*cos(self.inc*x)
```

```
self.w3=self.ong/self.inc*sin(self.inc*x)-cos(self.inc*x)*self.ina/(self.inc^2)

''')
```

E26A36_Antidiferentiation_Imediata_008_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas



Palavras-chave: Primitiva; antiderivada; tangente.


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas: tangente de x


Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"}:



<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re4+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re5+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>
```

```
%ANSWER

Como $\tan \; x = \frac{\sin x}{\cos x}$ e  $(\cos x)'= - \sin x$, podemos escrever,
\begin{eqnarray*}
\int f1 \, dx &=& sgn_a1 aa1 \int \, \frac {- \sin(x)}{\cos(x)}  \, dx\\
\\
&=&  resposta0
\end{eqnarray*}
porque $\big(\ln u \big)'= \frac{u'}{u}$.
Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portant
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.




class E26A36_Antidiferentiation_Imediata_008_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0,-1,1]) # parametro a multiplicar: de -9 a 9, exceto 0,-

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)




s.f1=s.a1*tan(x)
s.resposta0 =  - s.a1 * ln(abs(cos(x)))
```

```
#respostas
#s.re1 = diff(s.f1, x)              # derivada     ------------------------------------
s.re1 = s.a1/(cos(x))^2
s.re2 = s.a1 * ln(abs(cos(x)))     # esquece que a derivada do cosseno é o simétrico do
s.re3 = s.a1/2 * tan (x) * tan(x)  # primitiva a tangente como potência de exp 1 sem ter
s.re4 = s.a1 * ln(abs(sin(x)))     # def. tan errada
s.re5 = -s.re4                      # simetrico para desfazer o padrão de reposta

# Controlar os Sinais
if s.a1>0:
s.sgn_a1='-'
else:
s.sgn_a1=' '
s.aa1= abs(s.a1)


#função para reescrever   potências de funções trigonométricas e log-> ln
def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\((([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\cos\\left\((([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\cos^2(\1)'  ),
( ur'\\tan\\left\((([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\\tan^2(\1)'  ),
( ur'\\tan'                                         ,   r'{tg}'    ),
( ur'\\log'                                         ,   r'\\ln'  )
]

#Esta parte é sempre a mesma.
```

```
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text


''')
```

E26A36_Antidiferentiation_Imediata_009_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas



Palavras-chave: Primitiva; antiderivada; logaritmo.


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas:logaritmo de x


Seja  a função definida em
$] zerodenom ,   +\infty,[\;\;\ $
por  $$ f(x)= f1.$$
Uma primitiva desta função é:




<multiplechoice>
<choice> $$ g(x) = resposta0  $$ </choice>
```

```
<choice> $$ g(x) = re1 $$ </choice>
<choice> $$ g(x) = re2$$ </choice>
<choice> $$ g(x) = re3 $$ </choice>
</multiplechoice>



%ANSWER


<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula e


<showone res0>
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
Assim, derivando a função $g(x) = resposta0  $,
obtém-se $$g'(x) = f1, $$ portanto $g(x) = resposta0   $ é uma   primitiva  de $f(x) = f
</thisone>
<thisone Caso 1 - (isto é comentário)>
Uma primitiva desta função pode ser obtida tendo em conta que $(\ln (u) )' = \frac{u'}{u

\begin{eqnarray*}
\int f1 \, dx &=& aux1 \int \, aux2  \; dx\\
\\
&=&  resposta0.
\end{eqnarray*}
</thisone>
</showone>


class E26A36_Antidiferentiation_Imediata_009_siacua(ExSiacua):


def make_random(s,edict=None):
```

```
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a3 = ur.iunif_nonset(-9,8,[-1,0]) # parametro a multiplicar: de -9 a 9, exceto 0 e -
s.b1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.c1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

if gcd(abs(s.a2), abs(s.a3))>1:
s.a3 =s.a3+1

s.f1=s.a1/(s.a2*x + s.a3)
s.denom = s.a2*x + s.a3
s.zerodenom = - s.a3/s.a2

s.mdc=gcd(gcd(abs(s.a1), abs(s.a2)), abs(s.a3))  #máximo divisisor comum dos 3 paramet
# (isto é importante porque se o numerador e denominador da função f1 têm um factor co



if s.mdc > 1:
s.denom = s.denom/s.mdc

s.aux1 = s.a1/s.a2         #variável auxiliar para usar na resolução detalhada (por ca
s.aux2 = s.f1*s.a2/s.a1    #variável auxiliar para usar na resolução detalhada (por ca
#s.resposta0 =  s.aux1 * ln(abs(s.denom))

if s.a2 <0:
```

```
s.denom = -s.denom     # para escrever correctmente quando a2 é negativo
else:
s.dom0 = 1
s.resposta0 =  s.aux1 * ln(s.denom)

#print("a1=", s.a1)
#print("a2=", s.a2)
#print("a3=", s.a3)
#print("mdc=", s.mdc)
#respostas
s.re1 = diff(s.f1, x)                              # derivada-------------------------
s.re2 = 3*s.resposta0                              # mantém a constante do numerador--
s.re3 = s.re1  / s.a2                        # esquece de dividir pelo coeficiente do

#função para reescrever   potências de funções trigonométricas e log-> ln
def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\((([\d\\, u]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\log'  ,   r'\\ln'  )
]

#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text

''')
```

E26A36_Antidiferentiation_Imediata_010_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas



Palavras-chave: Primitiva; antiderivada; raiz de índice variável.


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas:raízes em denominador


Uma família de primitivas da função definida  por  <!-$ f(x)= f1$ ->
$f(x)= \frac{a1}{\sqrt[b1]{x}}$
sseuu@c{"é", "pode ser"}:



<multiplechoice>
<choice> $$ g(x) = aux0 \sqrt[b1]{x^{ aux1}} + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) =\frac{num0}{ b1 \sqrt[b1]{ x^{  exp0  } }}               + C, \; C
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>



%ANSWER


<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula
```

```
<showone res0>
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>

Assim, derivando qualquer função da família $g(x) = aux0 \sqrt[b1]{x^{ aux1}}  + C $, se
obtém-se $$g'(x) = \frac{a1}{\sqrt[b1]{x}}, $$
portanto, $g(x) = aux0 \sqrt[b1]{x^{ aux1}}  + C $ é uma família de primitivas de $f(x)
</thisone>

<thisone Caso 1 - (isto é comentário)>
Uma vez que $ \frac{1}{\sqrt[b1]{x}} $ se pode escrever como  $x^{\frac{-1}{b1}}$, uma p
<p>
Assim,

\begin{eqnarray*}
\int \frac{a1}{\sqrt[b1]{x}} \, dx
&=& a1 \int   x^{\frac{-1}{b1}} \, dx \\
&=& a1  \frac{x^{ -\frac{1}{b1}+1}}{-\frac{1}{b1}+1}\\
&=& aux0  x^{ \frac{b1 - 1}{b1}} \\
&=& aux0 \sqrt[b1]{x^{ aux1}}
\end{eqnarray*}



Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portanto,
$$ g(x) = aux0 \sqrt[b1]{x^{ aux1}} + C, \; C \in \mathbb{R} $$ é uma   primitiva de $f$
</thisone>
</showone>

class E26A36_Antidiferentiation_Imediata_010_siacua(ExSiacua):
```

```
def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a3 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.b1 =  ur.iunif(3,9) # inteiro de 3 a 9 -- é o indice da raiz
s.c1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0


#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)



s.f1=s.a1*(x^(-1/s.b1))
s.resposta0 =  s.a1*s.b1 /(s.b1 - 1)*x^((s.b1 - 1)/(s.b1))
s.aux0 = s.a1*s.b1/ (s.b1-1)
s.aux1 =  s.b1-1
s.num0 =  s.a1*(1+s.b1)
s.exp0 =  1+s.b1


#respostas
s.re1 = diff(s.f1, x)                                # derivada----------------------
s.re2 = s.a1*(1+s.b1) / (s.b1*x^((1+s.b1)/s.b1))     # primitiva o polinómio do denomi
s.re3 = ((s.a1*s.b1)/(s.b1+1))*x^((1+s.b1)/(s.b1))   # primitiva a função como se esta


''')
```

E26A36_Antidiferentiation_Imediata_011_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas



Palavras-chave: Primitiva; antiderivada; raiz.


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas: raiz de x


Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"}:



<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>



%ANSWER


<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula e


<showone res0>
```

```
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um nú
obtém-se $$g'(x) = f1, $$ portanto $g(x) = resposta0  + C $ é uma família de primitiva
</thisone>
<thisone Caso 1 - (isto é comentário)>
Como $ \frac{1}{\sqrt{x}} $ se pode escrever como  $x^{\frac{-1}{2}}$, uma primitiva d




Assim,
\begin{eqnarray*}
\int \frac {a1}{\sqrt {x}}\, dx &=& a1 \int \, x^{-\frac {1}{2}} \, dx\\
\\
&=& a1 \, \frac {x^{\frac {1}{2}}}{\frac {1}{2}} \\
\\
&=& 2\times (a1) \times x^{\frac {1}{2}} \\
\\
&=&  resposta0.
\end{eqnarray*}
Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portant
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
</thisone>
</showone>




class E26A36_Antidiferentiation_Imediata_011_siacua(ExSiacua):


def make_random(s,edict=None):
```

```
x=var('x')


s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0


#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)


s.exp1 =2


s.f1=s.a1/sqrt(x)
s.resposta0 =  2*s.a1*sqrt(x)



#respostas
s.re1 = diff(s.f1, x)            # derivada     ----------------------------------------
s.re2 = 2/3 * s.a1 * x^(2/3)     # calcula a primitiva de a1 x^{1/2} em vex de a1 x^{-1/
s.re3 = 1/2*s.resposta0          # esquece a divisão pelo expoente de x. (1/2) ---------




''')
```

E26A36_Antidiferentiation_Imediata_012_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas



Palavras-chave: Primitiva; antiderivada; potências de funções quadráticas.


Autores: João Silva
```

```
Ano: 2017

Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)

%problem Calcular primitivas imediatas: potências de funções quadráticas de x

Uma família de primitivas da função definida  por  $$ f(x)= f1 $$ sseuu@c{"é", "pode s


<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re4+ C, \; C \in \mathbb{R} $$ </choice>

</multiplechoice>



%ANSWER

<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula

<showone res0>
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um núm
obtém-se $$g'(x) = f1, $$ portanto, $g(x) = resposta0  + C $ é uma família de primitiv
</thisone>
<thisone Caso 1 - (isto é comentário)>
```

```
Uma primitiva desta função pode ser obtida usando a regra de primitivação da potência.

Observe-se que  $ 2x $ é a derivada da base da potência $(base0)^{b1}$.

Assim,
\begin{eqnarray*}
\int f1 \, dx &=& a11 \int \, 2x(x^2 sgn_a2 a22)^{b1} \, dx\\
&=& \frac{a12}{(b1 + 1)}(x^2 sgn_a2 a22)^{b1 + 1} \, dx\\
&=&  resposta0
\end{eqnarray*}

Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portanto,
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
</thisone>
</showone>



class E26A36_Antidiferentiation_Imediata_012_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0,2]) # parametro a multiplicar: de -9 a 9, exceto 0,1, -1
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a3 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.b1 = ur.iunif_nonset(1,9,[1]) # inteiro de 2 a 9
s.c1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
```

```
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

s.f1  =s.a1*x*((x^2 + s.a2)^(s.b1))
if s.a1 == 2:
s.a11 = ' '
s.a12 = ' 1 '
elif s.a1 == -2:
s.a11 = '-'
s.a12 = '-1 '
else:
s.a11 = s.a1/2
s.a12 =s.a11


#s.a11 = s.a1/2
s.resposta0=s.a1/(2*(s.b1+1))*(x^2 +s.a2)^(s.b1+1)
s.base0 = x^2 + s.a2




#respostas
s.re1 = diff(s.f1, x)                    # derivada     ------------------------------
s.re2 = 2*s.resposta0                    # 2/3 * s.a1 * x^(2/3)calcula a primitiva de
s.re3 = (s.b1 + 1)*s.resposta0           # 1/2*s.resposta0esquece a divisão pelo expo
s.re4 = s.a1*x^2*((x^2 + s.a2)^(s.b1+1))


# Controlar os Sinais
```

```
if s.a2<0:
s.sgn_a2='-'
else:
s.sgn_a2='+'
s.a22= abs(s.a2)

''')
```

E26A36_Antidiferentiation_Imediata_013_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas



Palavras-chave: Primitiva; antiderivada; raiz.


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas: raiz de (1-a x^2) em denominador


Uma família de primitivas da função definida  por  $ f(x)= f1$  sseuu@c{"é","pode ser"}:




<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
```

```
</multiplechoice>



%ANSWER

<!-- Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regu

<showone res0>
<thisone Caso 0>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.<br/>

Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um núm
obtém-se $$g'(x) = f1, $$ portanto $g(x) = resposta0  + C $ é uma família de primitiva
</thisone>
<thisone Caso 1>

Uma primitiva desta função pode ser obtida usando a regra de primitivação da potência
Observe-se que $\sqrt{1 sgn1_a2 a22 x^2}$ se pode escrever $(1 sgn1_a2 a22 x^2)^{\frac
e que $derbase1$ é a derivada da base desta potência.<br/>



Assim,
\begin{eqnarray*}
\int f1 \, dx
a1\, \int  x \, ( sgn_a2 a22 x^2 +1)^{-\frac{1}{2}} \, dx\\
\frac{a1}{ sgn_a2 2 \times  a22} \, \int \, (sgn_a2 2 \times a22)  x\, (sgn_a2  a22  x
aux0 \frac{(sgn_a2 a22 x^2 +1)^{\frac{1}{2}}}{\frac{1}{2}} \\
resposta0.
\end{eqnarray*}

Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portant
```

```
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.

</thisone>
</showone>




class E26A36_Antidiferentiation_Imediata_013_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu = ur.iunif(0,1) #escolhe aleatoriamente a expressao a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0,1]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opcao lista
s.res0   = ur.iunif(0,1) #opcao de escrita da resolucao 0,1  (2 possibilidades)

print "Controlos:", s.sseuu, s.a1, s.a2, s.res0



s.f1 =(s.a1*x)/ sqrt(1 - s.a2*x^2)
#s.resposta0 =  s.a1/s.a2 *sqrt(1-s.a2*x^2)
s.resposta0 =  integrate(s.f1,x)
s.aux0 =  - s.a1/(2*s.a2)
s.aux1 = 2* s.aux0
s.derbase1 = -2* s.a2*x

# Controlar os Sinais
if s.a2>0:
```

```
s.sgn_a2='-'
s.sgn1_a2='-'
else:
s.sgn_a2=' '
s.sgn1_a2='+'
s.a22= abs(s.a2)


#respostas
s.re1 = diff(s.f1, x)                           # derivada
s.re2 = s.a1 / (s.a2 * sqrt(1 - s.a2*x^2) )    # confunde/troca o expoente 1/2 por -1/
s.re3 = 2*s.a1 * sqrt(1-s.a2*x^2)              # esquece a divisao por 2*a2


''')
```

E26A36_Antidiferentiation_Imediata_014_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas


Palavras-chave: Primitiva; antiderivada;


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas: função linear com expoente negativo -- é um ca


Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"
```

```
<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>


%ANSWER


<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula e


<showone res0>
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um númer
obtém-se $$g'(x) = f1, $$ portanto $g(x) = resposta0  + C $ é uma família de primitivas
</thisone>
<thisone Caso 1 - (isto é comentário)>
Uma primitiva desta função pode ser obtida usando a regra de primitivação das potências


Assim,
\begin{eqnarray*}
\int \frac{a1}{\sqrt[b1]{aux11}} \, dx &=& \int \, a1 (aux11)^{-\frac{1}{b1}} \, dx\\
&=& aux21  \, \int \, a1 \times aux11l aux11d aux11r (aux11)^{-\frac{1}{b1}} \, dx \\
&=& aux31 \int \, aux11d (aux11)^{-\frac{1}{b1}} \, dx \\
&=& aux31 \times \frac{(aux11)^{1-\frac{1}{b1}}}{1-\frac{1}{b1}} \, dx \\
```

```
&=&  resposta0.
\end{eqnarray*}

Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portant
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
</thisone>
</showone>




class E26A36_Antidiferentiation_Imediata_014_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a3 = ur.iunif_nonset(-9,9,[0,-1,1]) # parametro a multiplicar: de -9 a 9, exceto 0
s.b1 =  ur.iunif(3,9) # inteiro de 3 a 9 -- é o indice da raiz

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

s.f1=s.a1/(s.a2-s.a3*x)^(1/s.b1)
#s.resposta0=s.a1/(-s.a3*(-1/s.b1+1))*(s.a2-s.a3*x)^(-1/s.b1+1)
s.resposta0 =  integrate(s.f1,x)
s.aux0 =  - s.a1/(2*s.a2)
s.aux1 = 2* s.aux0
#variável auxiliar que escreve corretamente esta expressao:a2-a3x
s.aux11=s.a2-s.a3*x
```

```
#derivada para usar nos cálculos
s.aux11d=derivative(s.aux11,x)
if s.a3<0:
s.aux11l=''
s.aux11r=''
else:
s.aux11l='('
s.aux11r=')'
s.aux21=1/s.aux11d
s.aux31=s.a1*s.aux21

# Controlar os Sinais
if s.a2>0:
s.sgn_a2='-'
else:
s.sgn_a2=' '
s.a22= abs(s.a2)


#respostas
s.re1 = diff(s.f1, x)        # derivada    ----------------------------------------------
s.re2 = s.a1/(-s.a3*(-s.b1+1))*(s.a2-s.a3*x)^(-s.b1+1)   # confunde/troca o expoente -1/
s.re3 = (1-1/s.b1) * s.resposta0          # esquece a divisão por -1/b1 + 1 -----------




''')
```

E26A36_Antidiferentiation_Imediata_015_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas
```

```
Palavras-chave: Primitiva; antiderivada; exponencial.

Autores: João Silva

Ano: 2017

Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)

%problem Calcular primitivas imediatas: exponencial.

Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser


<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re4+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>


%ANSWER

<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regul

<showone res0>
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
```

```
Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um númer
obtém-se $$g'(x) = f1, $$ portanto, $g(x) = resposta0  + C $ é uma família de primitivas
</thisone>

<thisone Caso 1 - (isto é comentário)>
Uma primitiva desta função pode ser obtida usando a regra de primitivação da função expo
<p>
Neste caso, $ u= e0 $  e  $u'= de0$.
<p>
Assim,
\begin{eqnarray*}
\int f1 \, dx &=& \frac{1}{a2} \, \int  f2 \, dx \\
&=&  resposta0.
\end{eqnarray*}
Portanto, qualquer função do tipo
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
</thisone>
</showone>



class E26A36_Antidiferentiation_Imediata_015_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(0,9,[0,1]) # parametro a multiplicar: de -9 a 9, exceto 0 e 1
```

```
#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

s.lista0=[s.a2*x + s.a1,  cos(s.a2*x), sin(s.a2*x), s.a2*x^2] #as   funções geradas pa
s.id3=ZZ.random_element(len(s.lista0 )) # gerado aleatoriamente entre 0 e o compriment
s.e0  = s.lista0[s.id3]
s.f1  =  exp(s.e0  )*diff(s.e0, x)/s.a2    #  função escolhida. A primitiva é imediata
s.f2  =  s.f1 * s.a2
s.resposta0 = integrate(s.f1,x)



#respostas
s.re1 = diff(s.f1, x)               # derivada     ----------------------------------
s.re2 = e^(s.e0)                    # primitiva como se fosse e^x (invariante)      --
s.re3 = 1/(s.a1*x + s.a2 + 1)*e^(s.a1*x + s.a2 + 1)          # primitiva como se e^x i
s.re4 = e^(s.e0) * integrate(diff(s.e0, x)/s.a2,x)

# Controlar os Sinais
if s.a2>0:
s.sgn_a2='+'
else:
s.sgn_a2='-'
s.aa2= abs(s.a2)

#função para reescrever exponenciais sem parentesis

def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'e\^\{\\left\((.+?)\\right\)\}'              ,   ur'e^{\1}'       ),
```

```
( ur'\\sin\\left\(([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\cos\\left\(([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\cos^2(\1)'  ),

]

#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text



''')
```

E26A36_Antidiferentiation_Imediata_016_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas


Palavras-chave: Primitiva; antiderivada; exponencial.

Autores: João Silva

Ano: 2017

Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)

%problem Calcular primitivas imediatas: exponencial -- absorvido pelo E26A36_Antidiferen

Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"}:
```

```
<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>


%ANSWER

<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula

<showone res0>
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um núm
obtém-se $$g'(x) = f1, $$ portanto $g(x) = resposta0  + C $ é uma família de primitiva
</thisone>
<thisone Caso 1 - (isto é comentário)>
Uma primitiva desta função pode ser obtida usando a regra de primitivação da função ex
<p>


Assim,
\begin{eqnarray*}
\int a1 x e^{sgn_a2 a22 x^2} \, dx &=& \frac{a1}{ sgn_a2  a22  \times 2 \,} \, \int  s
\\
&=& \frac{a1}{sgn_a2  a22  \times 2 \,}  \; e^{sgn_a2 a22 \, x^2} \\
\\
```

```
&=&  resposta0.
\end{eqnarray*}
Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portanto,
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
</thisone>
</showone>




class E26A36_Antidiferentiation_Imediata_016_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0,1]) # parametro a multiplicar: de -9 a 9, exceto 0,1
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a3 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1,0,
s.b1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1,0,
#s.a4 = -(s.a2)*x^2

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

#print("a1=", s.a1)
#print("a2=", s.a2)



s.f1 = s.a1*x*e^(s.a2*x^2)
```

```
#s.resposta0 =  -s.a1/(2*(s.a2))*exp(-(s.a2)*x^2)
s.resposta0 =  integrate(s.f1,x)
s.aux0 =  - s.a1/(2*(s.a2))*exp(-(s.a2)*x^2)
#s.aux1 = 2* s.aux0

# Controlar os Sinais
if s.a2>0:
s.sgn_a2=' '
else:
s.sgn_a2='-'
s.a22= abs(s.a2)



#respostas
s.re1 = diff(s.f1, x)            # derivada     ---------------------------------------

s.re2 = s.a1*exp(-(s.a2)*x^2)                    # primitiva como se fosse e^x (invar

s.re3 = -s.a1/(s.a2)*exp(-(s.a2)*x^2)          # primitiva como se e^x fosse função p

#função para reescrever exponenciais sem parentesis

def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'e\^\{\\left\((.+?)\\right\)\}'                ,   ur'e^{\1}'       )

]
```

```
#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text
```

```
''')
```

E26A36_Antidiferentiation_Imediata_017_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas


Palavras-chave: Primitiva; antiderivada; exponencial de base a diferente de e.

Autores: João Silva

Ano: 2017

Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)

%problem Calcular primitivas imediatas: exponencial de base a.

Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"}:
```

```
<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>


%ANSWER


Atendendo a que, para todo o número real $a$ positivo e diferente de $1$ se tem  $\le
$$ \int a^u \; u’ \;dx = \frac{a^u}{\ln (a)},$$ e usar esta fórmula para calcular uma
<p>


Assim,
\begin{eqnarray*}
\int f1   \, dx &=& \frac{1}{a2 \, \ln (b1)} \, \int a2 \, \times \,  b1^{a2 x - sgn_a
&=&  resposta0.
\end{eqnarray*}
Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portant
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.




class E26A36_Antidiferentiation_Imediata_017_siacua(ExSiacua):

def make_random(s,edict=None):
x=var(’x’)
```

```
s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a3 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1,0,
s.b1 = ur.iunif (2,9 ) # base a

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

s.f1=s.b1^(s.a2*x - (s.a3))



#s.resposta0 =  (-s.a1/(s.a2*ln(s.b1))) * s.b1^(-s.a2*x - (s.a3))
s.resposta0 = integrate(s.f1,x)

#respostas
s.re1 = diff(s.f1, x)           # derivada     ----------------------------------------

s.re2 = ln(s.b1)* s.resposta0                   # esquece de dividir por ln da base

s.re3 = s.a2* s.resposta0         # esquece de dividir pela derivada do expoente -------



# Controlar os Sinais
if s.a2<0:
s.sgn_a2='-'
else:
s.sgn_a2='+'
s.a22= abs(s.a2)
```

```
if s.a3<0:
s.sgn_a3='+'
else:
s.sgn_a3=' '
s.a33= abs(s.a3)

#função para reescrever  potências de funções trigonométricas e log-> ln
def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\((([\d\\, u]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\log'  ,   r'\\ln'  )
]

#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text



''')
```

E26A36_Antidiferentiation_Imediata_018_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas


Palavras-chave: Primitiva; antiderivada; exponencial.
```

```
Autores: João Silva

Ano: 2017

Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)

%problem Calcular primitivas imediatas: logaritmo; Primitiva   imediata

Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"}:


<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>


%ANSWER

Como  $ \left(\ln u \right)'= \frac{u'}{u}$, tem-se que
$$\int \frac{u'}{u} = \ln u$$
onde $u$ é uma função de $x$ tal que $\ln u$ (o logaritmo natural de $u$) é uma função q
<p>
Podemos usar esta fórmula:
<p>

\begin{eqnarray*}
\int f1 \, dx  &=&  resposta0
\end{eqnarray*}
```

```
e, portanto, qualquer função do tipo
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.




class E26A36_Antidiferentiation_Imediata_018_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif(0,9) # parametro a multiplicar: de -9 a 9,
s.a2 = ur.iunif(2,9) # parametro a multiplicar
s.n0 = ur.iunif(2,5)
s.b1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1



#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

s.lista0=[(s.a2*x + s.a1)^2,    s.a2*x^2+ s.a1,    x^s.n0+s.a2] #as   funções geradas
s.id3=ZZ.random_element(len(s.lista0 )) # gerado aleatoriamente entre 0 e o compriment
s.e0  = s.lista0[s.id3]
#print('e0=',s.e0)
s.f1  =   diff(s.e0, x)/s.e0   #  função escolhida. A primitiva é imediata: logaritmo
s.de0 = diff(s.e0,x)
s.resposta0 = integrate(s.f1,x)
#print('de0=', diff(s.e0, x))
```

```
#respostas
s.re1 = diff(s.f1,x)                                # derivada     ------------------------
s.re2 = 1/ s.resposta0                              #       ----------->
s.re3 =  integrate(s.de0,x)/integrate(s.e0,x)                             #  ------------>

if s.a1 * s.a2 <0:
s.sgn_p1='-'
else:
s.sgn_p1=' '
s.a11= abs(s.a1)
s.a22= abs(s.a2)

#função para reescrever   potências de funções trigonométricas e log-> ln
def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\((([\d\\, u]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\log'  ,   r'\\ln'  )
]

#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text

''')
```

E26A36_Antidiferentiation_Imediata_019_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas



Palavras-chave: Primitiva; antiderivada; exponencial


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas: exponencial ???? NÃO VERIFICADO - COMPLICADO!


Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"



<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>



%ANSWER


<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula


<showone res0>
<thisone Caso 0 - (isto é comentário)>
```

```
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um númer
obtém-se $$g'(x) = f1, $$ portanto $g(x) = resposta0  + C $ é uma família de primitivas
</thisone>
<thisone Caso 1 - (isto é comentário)>
Uma primitiva desta função pode ser obtida usando a regra de primitivação da função expo
<p>



Assim,
\begin{eqnarray*}
\int \frac {(ln x)^{a1}}{a2 x} \, dx &=& \frac{1}{a2}\int \frac{1}{x} \times (ln x)^{a1}
\\
&=& \frac{a1}{a2} \, \frac {(ln x)^{a1 + 1}}{a1 + 1} =\\
\\                                \\
&=&  resposta0.
\end{eqnarray*}
Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portanto,
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
</thisone>
</showone>




class E26A36_Antidiferentiation_Imediata_019_siacua(ExSiacua):


def make_random(s,edict=None):
x=var('x')


s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
```

```
s.a1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a3 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1
s.b1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)




s.f1=((ln (x)^(s.a1)))/(s.a2 * x)



s.resposta0 =integrate(s.f1,x)



#respostas
s.re1 = diff(s.f1, x)            # derivada     ----------------------------------------

s.re2 = (s.a1 + 1)* s.resposta0                  # esquece de dividir pelo expoente +

s.re3 = (s.a1 + 1)/s.a2*(ln(x))^(s.a1 + 1)         # Multiplica pelo expoente + 1 em v
#função para reescrever   potências de funções trigonométricas e log-> ln
def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\((([\d\\, u]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\log'  ,   r'\\ln'  )
```

```
]

#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text
''')
```

E26A36_Antidiferentiation_Imediata_020_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas



Palavras-chave: Primitiva; antiderivada;logaritmo.


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas: logaritmo --NÃO USAR  É CASO PARTICULAR DO E26A3


Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"}:



<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
```

```
<choice> $$ g(x) = re4+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>


%ANSWER

<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula

<showone res0>
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um núm
obtém-se $$g'(x) = f1, $$ portanto $g(x) = resposta0  + C $ é uma família de primitiva
</thisone>
<thisone Caso 1 - (isto é comentário)>
Uma primitiva desta função pode ser obtida usando a regra de primitivação do logaritmo
<p>



Assim,
\begin{eqnarray*}
\int f1 \, dx &=& aux0 \, \int \frac {\sin \, x}{\cos \, x} \, dx \\
&=& resposta0.
\end{eqnarray*}
Portanto, qualquer função do tipo
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
</thisone>
</showone>
```

```
class E26A36_Antidiferentiation_Imediata_020_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a3 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1,0,
s.b1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1,0,
s.a4 = -(s.a2)*x^2

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

s.f1 = (s.a1*sin(x)) / (s.a2*cos(x))




s.resposta0 =  integrate(s.f1,x)
s.aux0 =  s.a1/s.a2
#s.aux1 = 2* s.aux0



#respostas
s.re1 = diff(s.f1, x)                    # derivada     --------------------------------
s.re2 = s.a1/s.a2*tan(x)                      # esquece de dividir por a2      --------
s.re3 = s.a1/(2*s.a2)*(sin(x))^2         # primitiva o seno como se o cosseno estivesse
s.re4 = integrate(1/s.f1,x)
```

```
# Controlar os Sinais
if s.a2>0:
s.sgn_a2='-'
else:
s.sgn_a2=' '

if s.a1 * s.a2 <0:
s.sgn_p1='-'
else:
s.sgn_p1=' '

if s.sgn_p1 <0:
s.sgn_p2=' '
else:
s.sgn_p2='-'

s.a11= abs(s.a1)
s.a22= abs(s.a2)
s.a33= abs(s.a3)




#função para reescrever   potências de funções trigonométricas e log-> ln
def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\(([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\cos\\left\(([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\cos^2(\1)'  ),
```

```
( ur'\\tan\\left\(([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\\tan^2(\1)'  ),
( ur'\\tan'                                        ,   r'{tg}'    ),
( ur'\\log'  ,   r'\\ln'  )
]

#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text

''')
```

E26A36_Antidiferentiation_Imediata_021_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas imediatas



Palavras-chave: Primitiva; antiderivada; logaritmo.


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação é a operação inversa da derivação (Imediatas)


%problem Calcular primitivas imediatas: logaritmo


Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"}:



<multiplechoice>
```

```
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re4+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>


%ANSWER

<!-Há dois tipos de resposta que aparecem aleatoriamente: res0 é a variável que regula

<showone res0>
<thisone Caso 0 - (isto é comentário)>
Uma  primitiva de uma função é outra função cuja derivada é a função dada.
<p>
Assim, derivando qualquer função da família $g(x) = resposta0  + C $, sendo $C$ um núm
obtém-se $$g'(x) = f1, $$ portanto $g(x) = resposta0  + C $ é uma família de primitiva
</thisone>
<thisone Caso 1 - (isto é comentário)>
Uma primitiva desta função pode ser obtida usando a regra de primitivação do logaritmo
<p>


Assim,
\begin{eqnarray*}
\int f1 \, dx &=&  aux0 \, \int  \frac {\sin (b1 x)}{\cos (b1 x)} \, dx \\
&=&  aux1 \int b1  \,\frac {\sin (b1 x)}{\cos (b1 x)} \\
&=&  resposta0
\end{eqnarray*}
donde se pode concluir que qualquer função do tipo
```

```
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.
</thisone>
</showone>




class E26A36_Antidiferentiation_Imediata_021_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif(1,9) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a3 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1,0,
s.b1 = ur.iunif(1,9) # parametro a multiplicar: de -9 a 9, exceto -1,0,1
s.a4 = -(s.a2)*x^2
s.p1 = s.a1 * s.b1 * s.a2
s.p2 = s.a1 * s.b1

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

s.exp1 =2




s.f1 = (s.a1*sin(s.b1*x)) / (s.a2*cos(s.b1*x))
```

```
s.resposta0 =  integrate(s.f1,x)
s.aux0 =  s.a1/s.a2
s.aux1 = - s.aux0/s.b1



#respostas
s.re1 = diff(s.f1, x)                          # derivada     ----------------------
s.re2 = s.a1/s.a2*tan(s.b1*x)                  # converte sen x / cos x em tg x e nã
s.re3 =  s.a1/(2*s.a2)*(sin(s.b1*x))^2                 # não divide pelo coeficiente
s.re4 = integrate(1/s.f1,x)                 # considera que (cos x)’ = sen x ---->




# Controlar os Sinais
if s.a2>0:
s.sgn_a2=’-’
else:
s.sgn_a2=’ ’

if s.a1 * s.b1 * s.a2 <0:
s.sgn_p1=’ ’
else:
s.sgn_p1=’-’

if s.a1 * s.b1 <0:
s.sgn_p2=’-’
else:
s.sgn_p2=’ ’
```

```
s.a11= abs(s.a1)
s.a22= abs(s.a2)
s.a33= abs(s.a3)
s.b11= abs(s.b1)
s.p11= abs(s.p1)
s.p22= abs(s.p2)



#função para reescrever   potências de funções trigonométricas e log-> ln
def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\((([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\cos\\left\((([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\cos^2(\1)'  ),
( ur'\\log'  ,   r'\\ln'  )
]

#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text

''')
```

E26A36_Antidiferentiation_Parts_024_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas Por Partes
```

Palavras-chave: Primitiva; antiderivada.

Autores: João Silva

Ano: 2017

Propósito didático: Primitivação por partes

%problem Calcular primitivas usando o método e primitivação por partes: a x*sin(b x),

Uma família de primitivas da função definida por $ f(x)= f1$ sseuu@c{"é", "pode ser"

```
<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re4+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re5+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>
```

%ANSWER

Uma primitiva desta função pode ser obtida utilizando a regra de primitivação por part

```
<p>
\[
\int{f'(x) \, g(x) \, dx}=f(x)\, g(x)- \int{f(x) \, g'(x)\,dx}
\]
<p>
onde

\[
\begin{array}{lllllll}
f'(x) &=& f1simp , & \qquad & f(x)  &=& prim1 \\
g(x)  &=& x ,      & \qquad & g'(x) &=& 1 \\
\end{array}
\]

Assim,
\begin{eqnarray*}
\int f1 \, dx
&=& a1 \, \int x \,  f1simp \, dx \\
&=& a1 \, \left[   \left(prim1 \right) \, x \, - \int  \left(prim1 \right) \times 1 \, d
&=& a1 \, \left[  prim3 \, sgn_id3  \;  coef00     \;f1simp \right]\\
&=&  resposta0.
\end{eqnarray*}
Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portanto,
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.




class E26A36_Antidiferentiation_Parts_024_siacua(ExSiacua):


def make_random(s,edict=None):
```

```
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(0,9,[0]) # parametro a multiplicar: de 1 a 9
s.a3 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1
s.b1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto -1
s.a4 = -(s.a2)*x^2
s.p1 = s.a1 * s.b1 * s.a2
s.p2 = s.a1 * s.b1

#s.theta1=ur.random_element([0, pi/6,pi/2, pi/3,pi/4, pi ]) #opção lista
s.res0   = ur.iunif(0,1) #opçao de escrita da resolução 0,1  (2 possibilidades)

s.lista0=[s.a1*x*sin(s.a2*x), s.a1*x*cos(s.a2*x)] #as   funções geradas
s.id3=ZZ.random_element(len(s.lista0 )) # gerado aleatoriamente entre 0 e o compriment


s.f1  = s.lista0[s.id3]     #  função escolhida
s.f1simp =s.f1/(s.a1*x)     # f'
s.prim1 = integrate(s.f1simp,x)
s.prim2 = integrate(s.prim1,x)

s.prim3 = s.prim1 *x
s.coef0 = s.prim2/s.f1simp

#s.resposta0 =  (s.a1/(s.a2))*((sin(s.a2*x))/s.a2 - x*cos(s.a2*x) )
s.resposta0 =  integrate(s.f1,x)
```

```
#respostas erradas
s.re1 = diff(s.f1, x)        # derivada     ---------------------------------------------
s.re2 =   (s.a1/s.a2)*((sin(s.a2*x))/s.a2-cos(s.a2*x))                  # u'v - P u'v
s.re3 =   s.a2*s.resposta0   # ao primitivar não divide pelo argumento das funções trigo
s.re4 =   (s.a1/s.a2)*(sin(s.a2*x)-cos(s.a2*x))   # Não divide por a2 ao primitivar Pu'v
s.re5 =   s.a2*s.re4         # ao primitivar não divide pelo argumento das funções trigo



# Controlar os Sinais
if s.id3 ==1:
s.sgn_id3='+'
else:
s.sgn_id3='-'

if s.coef0 <0:
s.sgn_coef0='-'
else:
s.sgn_coef0='+'
s.a11= abs(s.a1)
s.a22= abs(s.a2)
s.a33= abs(s.a3)
s.b11= abs(s.b1)
s.p11= abs(s.p1)
s.p22= abs(s.p2)
s.coef00= abs(s.coef0)




#função para reescrever   potências de funções trigonométricas e log-> ln
def rewrite(self,text0):
```

```
#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\(([\d\\, u]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\log'  ,   r'\\ln'  )
]

#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text

''')
```

E26A36_Antidiferentiation_Parts_025_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas Por Partes



Palavras-chave: Primitiva; antiderivada.


Autores: João Silva


Ano: 2017


Propósito didático: Primitivação por partes


%problem Calcular primitivas usando o método e primitivação por partes: a x^2 exp(s.pu


Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"
```

```
<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>


%ANSWER

Uma primitiva desta função pode ser obtida utilizando a regra de primitivação por partes

<p>
\[
\int{f'(x) \, g(x) \, dx}=f(x)\, g(x)- \int{f(x) \, g'(x)\,dx}
\]

<p>

onde

\[
\begin{array}{lllllll}
f'(x) &=& exp0,    & \qquad & f(x)  &=& prim1 \\
g(x)  &=& x^2 ,      & \qquad & g'(x) &=& 2x \\
\end{array}
\]

<p>
```

```
Assim,
\begin{eqnarray*}
\int f1 \, dx
&=& a1 \, \int x^2 \, exp0 \, dx \\
&=& a1 \, \left[   prim1 \times (x^2) - \int 2 x prim1 \, dx \right] \\
&=& aux0  \; exp0  \, sgn1 aux11  \int x  exp0 \, dx
\end{eqnarray*}
e, usando de novo integração por partes para calcular este novo integral, agora com
\[
\begin{array}{lllllll}
f'(x) &=& exp0,    & \qquad & f(x)  &=& prim1 \\
g(x)  &=& x  ,      & \qquad & g'(x) &=& 1  \\
\end{array}
\]
fica,
\begin{eqnarray*}
\int f1 \, dx
&=& aux0  ; exp0  \, sgn1 aux11 \left( \frac{x}{a2}  exp0 - \int   prim1 \, dx   \righ
&=& aux0  \; exp0  \, sgn1 aux22 x exp0   sgn3 aux33 exp0 \\
&=&  resposta0.
\end{eqnarray*}
Portanto, qualquer função do tipo
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.



class E26A36_Antidiferentiation_Parts_025_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')
```

```
s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar: de -9 a 9, exceto 0
s.a2 = ur.iunif_nonset(0,9,[0]) # parametro a multiplicar: de 1 a 9
if abs(s.a1*s.a2) > 15:
s.a2 = 15//abs(s.a1)
s.f0 = s.a2*x
s.exp0 =exp(s.f0)
s.f2 = s.a1*x^2
s.f1 = s.f2 *s.exp0  #função a integrar
s.resposta0 =  integrate(s.f1,x)
s.aux0 = s.a1/s.a2 *x^2
s.aux1 = 2*s.a1/s.a2
s.aux2 = 2*s.a1/s.a2^2
s.aux3 = s.aux2 /s.a2
s.prim1=integrate(s.exp0,x)

#respostas erradas
s.re1 = diff(s.f1, x)                                # derivada     --------------------
s.re2 =  integrate(x^2,x)*s.exp0 - integrate(integrate(x^2,x) * derivative(s.exp0) , x)
s.re3 =   s.a1*s.exp0*(x^2 - 2*x+2)              # ao primitivar não divide por a2 semp

# Controlar os Sinais
if s.a1 >0:
s.sgn1 = '-'
s.sgn3 = '+'
else:
s.sgn1 = '+'
s.sgn3 = '-'

s.aux11= abs(s.aux1)
s.aux22= abs(s.aux2)
```

```
s.aux33= abs(s.aux3)

''')
```

E26A36_Antidiferentiation_Parts_026_siacua

```
meg.save(r'''
%summary Primitivas; Primitivas Por Partes

Palavras-chave: Primitiva; antiderivada.
Autores: João Silva
Ano: 2017
Propósito didático: Primitivação por partes

%problem Calcular primitivas usando o método e primitivação por partes:a ln(b x)

Uma família de primitivas da função definida  por  $ f(x)= f1$ sseuu@c{"é", "pode ser"

<multiplechoice>
<choice> $$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re1+ C,  \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re2+ C, \; C \in \mathbb{R} $$ </choice>
<choice> $$ g(x) = re3+ C, \; C \in \mathbb{R} $$ </choice>
</multiplechoice>

%ANSWER

Uma primitiva desta função pode ser obtida utilizando a regra de primitivação por part
<p>
Assim,
\begin{eqnarray*}
\int f1 \, dx
&=& a1 \, \int 1 \, \times \, \ln(a2 x)\, dx \\
```

```
&=& a1 \, \left( x \, \ln (a2 x) - \int x \frac {a2}{a2 x} \, dx \right) \\
&=& a1 \, \left( x \, \ln (a2 x) - x \right)\\
&=& a1 \, x \, \left(\ln (a2 x) - 1 \right)\\
&=&  resposta0.
\end{eqnarray*}

Adicionando uma qualquer constante a esta função obtém-se ainda uma primitiva, portanto,
$$ g(x) = resposta0 + C, \; C \in \mathbb{R} $$ é uma primitiva de $f$.

class E26A36_Antidiferentiation_Parts_026_siacua(ExSiacua):

def make_random(s,edict=None):
x=var('x')

s.sseuu= ur.iunif(0,1) #escolhe aleatoriamente a expressão a escrever no enunciado
s.a1 = ur.iunif_nonset(-9,9,[0,1,-1]) # parametro a multiplicar
s.a2 = ur.iunif_nonset(0,9,[0,1,-1]) # parametro a multiplicar
s.f1 = s.a1*ln(s.a2*x) #função a integrar
s.resposta0 =  integrate(s.f1,x)

#respostas erradas
s.re1 = diff(s.f1, x)        # derivada     -----------------------------------------
s.re2 =   s.a1*x*(ln(s.a2*x - 1/(s.a2)))        #Faz (ln u)'=1/u em vez de u'/u por hábi
s.re3 =   s.a1/2 *(ln (s.a2*x))^2       # Primitiva ln(a2 x) como sendo a base de portên

# Controlar os Sinais
if s.a2 >1:
s.sgn_a2 = '-'
else:
s.sgn_a2 = '+'
```

```
s.a11= abs(s.a1)
s.a22= abs(s.a2)

#função para reescrever   potências de funções trigonométricas e log-> ln
def rewrite(self,text0):

#Esta parte depende do que queremos alterar.
modifications = [
( ur'\\sin\\left\(([\d\\, u]+?)\\right\)\^\{2\}'  ,   r'\sin^2(\1)'  ),
( ur'\\log\\left\(([\d\\, x]+?)\\right\)\^\{2\}'  ,   r'\\log^2(\1)'  ),
( ur'\\log'  ,   r'\\ln'  ),
]

#Esta parte é sempre a mesma.
text = text0
for m in modifications:
text =  re.sub( m[0], m[1], text, re.U )
return text

''')
```