

XI JORNADAS DE REDES DE INVESTIGACIÓN EN DOCENCIA UNIVERSITARIA

Retos de futuro en la enseñanza superior:
Docencia e investigación para alcanzar la excelencia académica



ISBN: 978-84-695-8104-9

XI JORNADES DE XARXES D'INVESTIGACIÓ EN DOCÈNCIA UNIVERSITÀRIA

Reptes de futur en l'ensenyament superior:
Docència i investigació per a aconseguir l'excel·lència acadèmica

Coordinadores

María Teresa Tortosa Ybáñez

José Daniel Álvarez Teruel

Neus Pellín Buades

© **Del texto: los autores**

© **De esta edición:**

Universidad de Alicante

Vicerrectorado de Estudios, Formación y Calidad

Instituto de Ciencias de la Educación (ICE)

ISBN: 978-84-695-8104-9

Revisión y maquetación: Neus Pellín Buades

La simultaneidad de ejecución de software matemático en el proceso de enseñanza-aprendizaje

A Pérez Carrió

Departamento de Matemática Aplicada

Universidad de Alicante

RESUMEN

Este trabajo es la continuación natural de la investigación detallada en las X jornadas de redes de investigación en docencia universitaria en la que se presentó la comunicación “El software matemático como herramienta de refuerzo en la adquisición de competencias”. El resultado del refuerzo mediante la repetición de ejercicios realizados en el aula incidió de manera notable en el rendimiento académico. Con el objetivo de optimizar resultados se inició un estudio sobre el proceso de enseñanza-aprendizaje en el que se pretendía obtener el máximo rendimiento de dicho refuerzo a instancias del software matemático. Así pues, este proceso ha proporcionado elementos de mejora en la interacción docente-discente, basados en la detección de errores o equivocaciones en trabajos realizados en horas de clase. La ejecución simultánea de software de forma presencial ha potenciado el perfeccionamiento de procedimientos en la adquisición de competencias, observándose un avance significativo en la erradicación de los mencionados errores debido a la interacción provocada por la actuación paralela profesor/a-alumno/a en la realización de nuevos ejercicios de software matemático añadidos a los de refuerzo.

Palabras clave: Matemáticas, software, simbolismo, equivocaciones, simultaneidad.

1. INTRODUCCIÓN

1.1 Problema/cuestión.

La utilidad del software en el proceso de adquisición de competencias en el área de matemática aplicada es incuestionable, sobre todo en los grados de ingeniería. En la titulación de Grado de Ingeniería de Edificación y, particularmente, en las asignaturas Fundamentos de Matemática Aplicada I y II, se utiliza el manipulador simbólico Maple para la realización de prácticas con ordenador. El espíritu de dichas prácticas ha sufrido modificaciones a lo largo de estos tres cursos desde la implantación de los grados en 2010. Inicialmente se pensó, equivocadamente, que el alumnado era capaz de “crear” contextos de trabajo mediante la programación habitual en la línea de entornos como C++, Pascal, Delphi, etc. El objetivo de esta experiencia era formar al estudiante en estrategias de búsqueda, reconocimiento y análisis de técnicas de resolución de problemas sobre la disciplina mayoritaria que implica la titulación, en el ámbito particular de la materia que nos ocupa. Fue un proyecto demasiado ambicioso, pues el alumnado carecía de la preparación previa necesaria para abordarlo con éxito. El plan de aprendizaje fue unidireccional ya que no hubo respuesta a la demanda del profesorado, es más, se propuso una lista de ejercicios de libre resoluciónⁱ, que sólo una alumnaⁱⁱ contestó en su totalidad. La dificultad principal no consistía en saber lo que se pedía sino en implementar el algoritmo conducente a su resolución.

Después de este fracasoⁱⁱⁱ en la tentativa de utilizar la programación pura y dura para contribuir al desarrollo de estrategias, técnicas y modelos utilizables tanto en la ingeniería en general como en la materia de matemáticas, decidimos olvidarnos de la misma y dedicarnos al manipulador simbólico y su entorno de comandos preparados para actividades coordinadas con las realizadas en el aula. Éste fue el inicio de un programa de enseñanza-aprendizaje, basado en el manejo del software matemático, diseñado por etapas:

- Etapa 1: El software matemático como herramienta de resolución de ejercicios similares a los realizados en el aula.
- Etapa 2: El software matemático como herramienta de refuerzo^{iv} en la adquisición de competencias.
- Etapa 3: La simultaneidad de ejecución de software matemático en el proceso de enseñanza-aprendizaje

Se observa una evolución en las diferentes etapas pues, en la primera, los ejercicios resueltos en el laboratorio de informática eran similares a los realizados en el aula. En la segunda se repetían los ejercicios del aula y en la tercera, objeto de este estudio, se han realizado ejercicios distintos. Como se vio en las jornadas del curso pasado, el refuerzo mediante la repetición de ejercicios mejoró notablemente el rendimiento académico en la parte del seminario^v teórico-práctico y las prácticas de problemas, sin embargo en los controles de prácticas con ordenador las calificaciones seguían siendo medias. Esto no entra en contradicción con que el resultado global fue bastante satisfactorio, pero precisa de un ajuste que otorgue a la propia herramienta la posibilidad de ser valorada desde dentro. Es decir, se ha utilizado el software matemático para resolver problemas y cuestiones de ámbito matemático, pero no se ha realizado ningún esfuerzo en la búsqueda de técnicas que permitan comprender y asimilar de forma óptima el manipulador simbólico. Resulta obvio que cada etapa potencia la anterior por lo que podemos entender que esta última, en la que se utilizan técnicas de simultaneidad de ejecución de software, dará pie a otras.

1.2 Revisión de la literatura.

Utilizar un manipulador simbólico como Maple [5] implica un dominio del lenguaje matemático y de la sintaxis adecuada que garantice la consecución de los objetivos planteados. La evaluación de este software como herramienta de trabajo en determinados campos de la matemática como el álgebra lineal [7] y el cálculo infinitesimal [6] es misión del profesorado docente e investigador, ya que la informática una baza fundamental en el trabajo en “modo inmersión” [1] como se decía en las jornadas de 2012 [4]. Recientes trabajos [3] muestran la importancia del lenguaje y de los “átomos significativos” en aras del aprendizaje desde dentro del manipulador. No estamos hablando de compilaciones mentales en tiempo real, sino de la base de toda cultura: La escritura inteligible [2].

1.3 Propósito.

El propósito de esta investigación es fomentar la comprensión del manipulador para utilizarlo correctamente como herramienta de resolución de ejercicios de álgebra lineal y cálculo infinitesimal, continuando la labor iniciada cursos atrás [4]. El objetivo fundamental es la consecución de una buena sintaxis mediante la detección de equivocaciones o errores en

la ejecución simultánea del programa informático para la resolución de ejercicios sin praxis de aula.

2. DESARROLLO DE LA CUESTIÓN PLANTEADA

2.1 Objetivos

Los objetivos generales y específicos de las asignaturas “Fundamentos de Matemática Aplicada I” y “Fundamentos de Matemática Aplicada II”, marcados, respectivamente, por el área de matemática aplicada y el profesorado que imparte las asignaturas, fueron detallados en las X jornadas [4], por lo que nos centraremos en los objetivos propuestos para este trabajo, además de los basados en el manejo de manipuladores simbólicos, que recordamos a continuación:

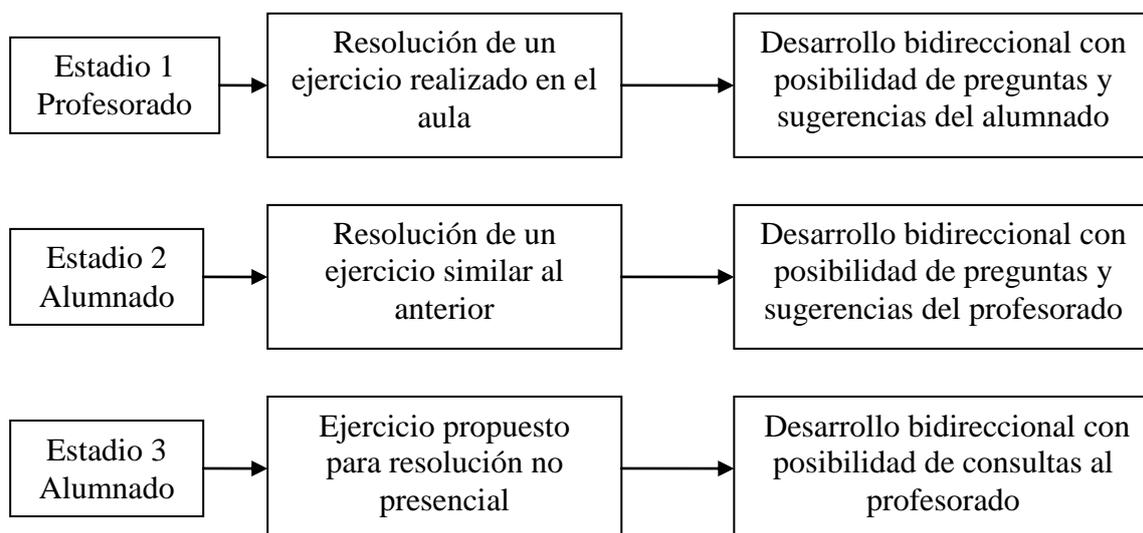
- Manejar con eficiencia los comandos inherentes a los conceptos matemáticos establecidos en el temario, dentro de los paquetes ofrecidos por el manipulador Maple 9.0.
- Utilizar en lo posible la simbología matemática usada en las clases de aula.
- Ejercitar el asentamiento de conceptos y procedimientos mediante el refuerzo de los mismos.
- Establecer el laboratorio de informática como el entorno de prácticas fuera del aula.
- Mejorar el rendimiento académico de los estudiantes en la disciplina de matemática aplicada.

Los nuevos objetivos añadidos a los anteriores son:

- Asimilar la sintaxis básica utilizada en las prácticas del laboratorio de informática.
- Saber distinguir entre error y equivocación.
- Detectar errores o equivocaciones con la suficiente agilidad.
- Saber corregir con éxito la sintaxis que induce mensajes de advertencia.
- Seguir el ritmo de resolución simultánea del profesor y del grupo.
- Tener autonomía para resolver problemas nuevos sin errores o equivocaciones

2.2. Método y proceso de investigación.

El proceso seguido en una clase de prácticas con ordenador venía dado por el siguiente esquema:



Este esquema es aparentemente correcto pues sigue un desarrollo bidireccional en el proceso de enseñanza-aprendizaje: El profesorado explica paso a paso un ejercicio que previamente ha sido resuelto en el seminario teórico-práctico o en prácticas de problemas. El “paso a paso” significa entrar en todos los detalles de sintaxis utilizados. Esto implica un esfuerzo de concentración bastante elevado que, como es habitual, no se consigue salvo en un porcentaje sumamente reducido del grupo. Como consecuencia, en el estadio 2, el alumnado intenta resolver un ejercicio similar, que siempre suele llevarse a cabo con la intervención del profesorado. Finalmente, en el tercer estadio, pocos estudiantes resuelven el ejercicio propuesto fuera del laboratorio y, además, las consultas son escasas. Sin embargo, aunque la intención es fomentar un desarrollo bidireccional, en realidad no lo es, pues la intensidad de la enseñanza no completa el aprendizaje.

Este método [4] implicó una mejora en el rendimiento de los seminarios y las prácticas de problemas pero no hubo la misma repercusión en los controles de las prácticas con ordenador. Es decir, la herramienta informática, siendo útil para el refuerzo, es infrutilizada. Lo que nos conduce a cambiar el esquema anterior y modificarlo a tenor de unos objetivos que, abundando en los anteriores, permitan conocer el manipulador en su fase de escritura,

con la sintaxis adecuada, de forma que la ejecución de un programa no quede paralizada por problemas que, en esencia, son básicos y, por lo tanto, subsanables.

En esta fase de la investigación se desarrolla un grupo de 20 cuestiones de autorreflexión^{vi} encaminado a encontrar los fallos que paralizan la ejecución del software.

- 1) ¿Conoces los comandos^{vii} que utilizas en Maple?
- 2) ¿Empleas la sintaxis aprendida al inicio de curso?
- 3) ¿Utilizas el comando **restart**^{viii} cuando tienes algún problema?
- 4) ¿Cargas el paquete^{ix} adecuado a los comandos usados?
- 5) ¿Cómo sabes si un paquete se ha cargado o no?
- 6) ¿Conoces la jerarquía de las operaciones básicas?
- 7) ¿Distingues lo que es un error de una equivocación o para ti son lo mismo?
- 8) ¿Entiendes los mensajes de advertencia?
- 9) ¿Comprendes los mensajes^x de error o equivocación?
- 10) ¿Después de un mensaje de error o equivocación, sabes en qué posición de la línea o en qué línea ha ocurrido?
- 11) ¿Cuando el resultado no es correcto, detectas dónde has fallado?
- 12) ¿Si el manipulador no funciona, sabes qué hacer?
- 13) ¿Si no sabes qué hacer, arrastras un error hasta el final?
- 14) ¿Consideras la lógica del resultado?
- 15) ¿Cómo utilizas un ejemplo como referencia?
- 16) ¿Conoces la forma de variar de modo texto a expresión matemática?
- 17) ¿Empleas secciones de título^{xi} y menús para distinguir las partes de un ejercicio?
- 18) ¿Haces anotaciones en modo texto para aclarar tus dudas o para poder revisar el ejercicio con garantías?
- 19) ¿Te has preocupado de mirar todos los iconos y menús del manipulador y conocer su utilidad?
- 20) ¿Utilizas la ayuda para ampliar tus conocimientos?

Dado que el cuestionario habla por sí mismo, sólo aclararemos las definiciones de algunos de los términos usados, que son clave en este proceso de investigación:

Sintaxis: Conjunto de reglas que definen las secuencias correctas de los elementos de un lenguaje de programación.

Error: Defecto en la sintaxis provocado por el desconocimiento de la misma.

Equivocación: Defecto en la sintaxis provocado por accidentes^{xii} de escritura.

Veamos en la siguiente imagen un error (no equivocación) común:

```
> f := (x, y) -> (1 - exp(x(y-2))) / (x(y-2));
```

$$f := (x, y) \rightarrow \frac{1 - e^{x(y-2)}}{x(y-2)}$$

Se ha definido una función de dos variables para estudiar su continuidad, posible ampliación de dominio y determinación de la función ampliada. Aparentemente todo está correcto pero si evaluamos la función en un punto que no anule el denominador...

```
> f(1, 12);
```

$$1 - e$$

Se observa que el resultado no tiene sentido. Pues bien, el defecto sintáctico está, claramente, en que el producto debe escribirse con *. Veamos ahora la sintaxis correcta:

```
> f := (x, y) -> (1 - exp(x*(y-2))) / (x*(y-2));
```

$$f := (x, y) \rightarrow \frac{1 - e^{x(y-2)}}{x(y-2)}$$

Aparentemente la función, en azul, es la misma que antes salvo que aparece un paréntesis en el exponente del numerador. Pero en esta ocasión, evaluándola en el punto, se obtiene el resultado correcto.

```
> f(1, 12);
```

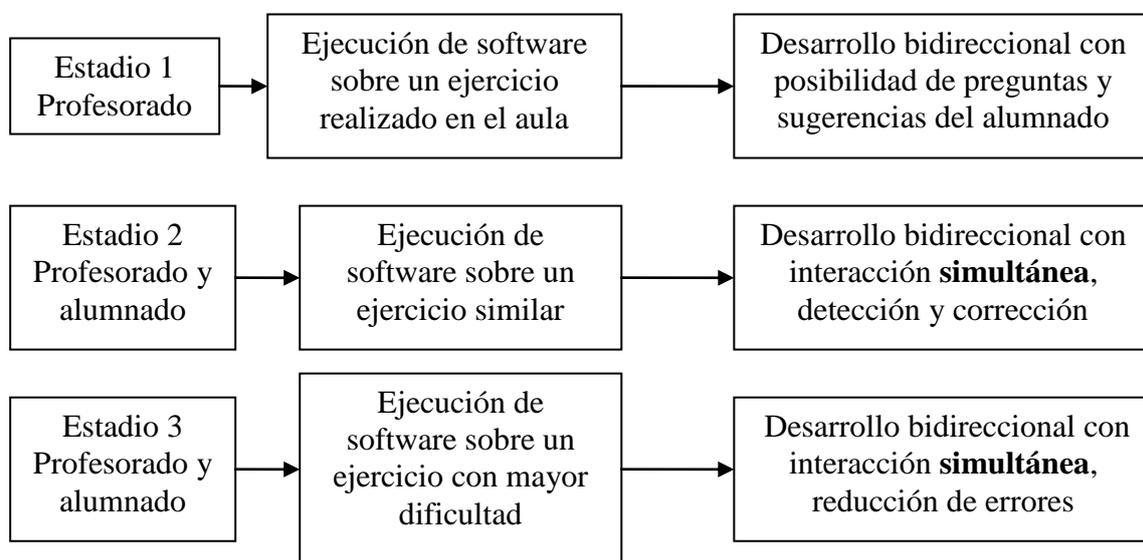
$$\frac{1}{10} - \frac{1}{10} e^{10}$$

Un error tan obvio como éste hace que el alumnado siga trabajando^{xiii} de forma incorrecta sin percatarse de que sólo está actuando la variable x.

Como podemos observar, el cuestionario indica el tipo de errores^{xiv} con el que habitualmente se tropieza al corregir una práctica o en la interacción en el laboratorio.

El siguiente paso es iniciar una metodología que implique la detección y rápida corrección de los posibles errores o equivocaciones. Una gran mayoría de alumnas y alumnos es incapaz, por desgracia, de llegar a conclusiones aceptables por sí mismos y, por tanto, precisa de una actuación, in situ, que le ayude en el momento de cometer el fallo. Esto se ha conseguido cambiando el esquema que se seguía en las clases de prácticas por ordenador. Recordemos que el alumnado tenía un papel protagonista en los estadios 2 y 3, mientras que el papel del profesorado es de supervisión, aclaración y corrección en un paso concreto de la ejecución de software, obviando el resto como actividad no presencial. Se trata de una solución local de los problemas pero no global como debería ser.

Este planteamiento nos lleva a elaborar una estrategia de **simultaneidad de ejecución**, en la que el profesorado y el alumnado resuelven el ejercicio al mismo tiempo partiendo de cero^{xv}. El esquema es, ahora, el siguiente:



Este nuevo esquema permite la colaboración del grupo, pues la interacción es global en contraposición al anterior, que era individual. Además, el ritmo de resolución queda marcado por los/las que tienen más dudas o problemas o, al menos, son capaces de decirlo, provocando un aumento de atención en el resto. Por otro lado se observa que la realización de

un segundo ejercicio, aunque tenga mayor dificultad^{xvi}, con las mismas condiciones de partida, hace que la detección y corrección de fallos sea más rápida. En definitiva, el trabajo simultáneo profesorado-alumnado genera una interacción que mejora ostensiblemente el proceso de enseñanza-aprendizaje. Hay que tener en cuenta que esta metodología es un reto para el profesorado, dado que tiene que generar una práctica al mismo tiempo que el alumnado, teniendo que enfrentarse en ocasiones a dificultades inesperadas^{xvii}, que permiten a éste ver cómo se solucionan dichos conflictos en tiempo real y acotado. También se produce un juego que podemos denominar “diagnóstico, tratamiento y cura según los síntomas”. Consiste en parar el proceso de resolución simultánea del ejercicio para atender las necesidades de algún alumno o alumna que tiene algún tipo de problema y no puede seguir el ritmo del resto. Entonces el/la alumno/a explica dónde se ha parado, en qué no coincide con los demás y finalmente cuáles son los síntomas^{xviii}. El diagnóstico está basado en la apariencia de los síntomas que, en la mayoría de los casos, son un referente exacto del problema. En ocasiones hay que recurrir a varias preguntas para poder deducir, sin lugar a dudas, el diagnóstico correcto. Lógicamente, el “tratamiento y cura” tienen una dificultad directamente proporcional a la gravedad del “diagnóstico”.

3. CONCLUSIONES

El principal objetivo de este trabajo ha sido potenciar la comprensión de la herramienta informática. Su uso como refuerzo para las clases en seminarios teórico-prácticos^{xix} y clases de problemas ha quedado probado en anteriores ediciones de estas jornadas. La forma de conseguir que el alumnado adquiriera conciencia de dicho instrumento es analizar los errores o equivocaciones detectados en procesos básicos como son: la propia sintaxis, el empleo de la lógica del resultado, la utilización de la ayuda, etc. La metodología utilizada en aras de un correcto manejo del manipulador simbólico se ha denominado proceso de interacción simultánea, que ha contribuido a la mejora de los siguientes ítems:

- a. Rapidez en la detección y corrección de fallos en la resolución de ejercicios.
- b. Participación del grupo en la resolución de conflictos en la ejecución de comandos en la realización de ejercicios..
- c. Desinhibición ante el resto del grupo.
- d. El proceso de enseñanza-aprendizaje.
- e. Los resultados académicos en las prácticas con ordenador.

Después de esta experiencia vamos a trabajar en la organización del juego de diagnóstico y el estudio de ritmos de trabajo para las siguientes jornadas. También tenemos la intención de migrar al entorno de software libre “Máxima”, con el fin de eliminar gastos debidos a los elevados, y prescindibles, costes de las licencias.

4. REFERENCIAS BIBLIOGRÁFICAS

- [1] Blyth, B., & Labovic, A. (2009). Using Maple to implement eLearning integrated with computer aided assessment. *International Journal of Mathematical Education in Science and Technology*, 40(7), 975-988.
- [2] Giandini, V. H., & Salerno, M. N. (2009). La Geometría, los Ingresantes y el Software Maple. *Formación Universitaria*, 2(4), 23-30.
- [3] Hannah, J., Stewart, S., & Thomas, M. (2013). Emphasizing language and visualization in teaching linear algebra. *International Journal of Mathematical Education in Science and Technology*, 44(4), 475-489.
- [4] Pérez, A., & Navarro, J.F. (2012). El software matemático como herramienta de refuerzo en la adquisición de competencias. *X Jornadas de Redes de Investigación en Docencia Universitaria*. ICE. Universidad de Alicante.
- [5] Pujol, G., Gibergans, J., & Buenestado, P. (2008). *Matemáticas para la ingeniería con Maple*. Barcelona: Edicions UPC.
- [6] Samková, L. (2012). Calculus of one and more variables with Maple. *International Journal of Mathematical Education in Science and Technology*, 43(2), 230-244.
- [7] Soto, M.J., & Vicente, J.L. (2001). *Álgebra lineal con Matlab y Maple*. Sevilla: Prentice Hall.

Notas

ⁱ Los ejercicios de libre resolución se pueden resolver en clase o fuera de ella sin necesidad de acotar el tiempo, el material o los medios. Este tipo de ejercicios presenta un sistema totalmente opuesto a los controles habituales que se realizan en el aula, en una fecha dada, con una duración limitada y con los materiales y medios disponibles en la misma.

ⁱⁱ La alumna provenía de la titulación de Arquitectura Técnica y después de abandonar los estudios durante unos años para dedicarse a la maternidad, regresó a la actividad discente trasladándose al Grado de Ingeniería de Edificación. La madurez de ésta en comparación con los estudiantes recién llegados a la titulación era notable.

ⁱⁱⁱ La observación de la falta de éxito en las tareas de programación tuvo lugar en el primer semestre del curso 2010-2011. De esta forma, en el segundo semestre del mismo curso, se optó por cambiar radicalmente la metodología de trabajo en el campo informático.

^{iv} Trabajo presentado en las X Jornadas de Redes de Investigación en Docencia Universitaria.

^v Las asignaturas “Fundamentos de matemática aplicada I y II”, de la titulación de Grado en Ingeniería de Edificación en la Escuela Politécnica Superior, son de 6 créditos ECTS, repartidos de la siguiente forma: Seminario teórico-práctico 3 créditos; prácticas de problemas 1.5 créditos; prácticas con ordenador 1.5 créditos.

^{vi} Estas cuestiones están preparadas para que el alumnado reflexione sobre lo que debería saber al inicio del segundo semestre. Las contestaciones son muy variopintas y en algunos casos se restringen a los consabidos monosílabos.

^{vii} Los comandos se identifican por su aplicación y por los parámetros que incluyen. En gran parte de los casos el conocimiento de un comando se restringe únicamente a su acción directa sin entrar en el estudio de los parámetros disponibles que podrían ofrecer información de utilidad.

^{viii} Se recomienda siempre que se busque la solución al problema y, si éste persiste, reiniciar el proceso junto con las variables. Cuando se realiza un ejercicio después de un ejemplo es lógico reiniciar las variables pues, aunque

la ejecución se hace con memoria sustitutiva, puede haber algún parámetro que enmascare los resultados al no haber suprimido su asignación previa.

^{ix} Cuando un paquete concreto no se carga y se utiliza un comando del mismo, éste no realiza ninguna acción y aparece escrito en cursiva junto al argumento sobre el que tenía que actuar. También puede aparecer un mensaje de color rosa advirtiendo que no se puede ejecutar el comando por la ausencia de dicho paquete.

^x Los mensajes de advertencia juegan un papel crucial en la detección de errores o equivocaciones. El color del mensaje marca la importancia del mismo. El color azul indica una advertencia sobre las propiedades del paquete, comando o función, o un aviso sobre la omisión del cierre de línea que no impide la ejecución de la misma. El color rosa advierte sobre la imposibilidad de ejecución de un comando o línea de programación y la causa de ésta.

^{xi} El sistema de evaluación de un control de prácticas con ordenador consta de tres parámetros: La presentación (1 punto), el desarrollo del ejercicio (8 puntos) y 5 cuestiones para determinar la comprensión de lo calculado en el ejercicio (1 punto).

^{xii} Con estas definiciones, una equivocación se convierte en error cuando los accidentes van acompañados de desconocimiento. Lógicamente, una equivocación es fácilmente detectable y subsanable, mientras que un error suele ser indetectable para el/la que lo comete debido, precisamente, a su falta de conocimiento sintáctico.

^{xiii} Debemos tener en cuenta que parte del alumnado no es capaz de admitir errores delante del grupo. Esto supone una traba individual que impide seguir el ritmo del resto, con perjuicio de su proceso de adquisición de competencias.

^{xiv} Son errores típicos: La ausencia de operadores y el desconocimiento de la jerarquía de los operadores básicos, la carga de paquetes improcedentes, la falta de carga de los mismos o el desconocimiento de la efectividad de la carga, la utilización de variables equivocadas, el cambio de denominación de una misma variable, las asignaciones incorrectas, los rangos fuera de campo, los comandos mal aplicados, los comandos y paquetes mal escritos, las funciones básicas mal escritas, los números notables mal escritos, la confusión entre mayúsculas y minúsculas, la falta de reinicio de las variables, las divisiones por cero y las condiciones imposibles.

^{xv} Habitualmente, y debido a que sólo tenemos una hora para explicar, desarrollar y resolver la práctica, se hace copia-pegar del ejemplo para después sustituir los valores nuevos y comprobar que todo funciona. Esto significa que partir de cero es no hacer copia-pegar y escribir toda la programación como si la práctica se estuviera creando en ese momento.

^{xvi} Significa que contempla más apartados con preguntas relacionadas con prácticas anteriores.

^{xvii} Las prácticas son entregadas, con suficiente antelación, por el coordinador de la asignatura, pero todos sabemos que cuando se maneja un manipulador simbólico en un ordenador o computador pueden darse situaciones no previstas.

^{xviii} Por ejemplo, cuando se tiene que calcular un determinante de una matriz numérica o variable, y en vez de hacerlo aparece la denominación del comando junto a la matriz. Ése es el síntoma. El diagnóstico es que no ha cargado el paquete **linalg**, el tratamiento es que debe escribir en la línea de comandos **with(linalg);** y la cura es ejecutar de nuevo la línea y comprobar que ahora se ejecuta el comando.

^{xix} Correspondientes a las materias Fundamentos de Matemática Aplicada I y II de la titulación de Grado en Ingeniería de Edificación de la Escuela Politécnica Superior de la Universidad de Alicante.