

Més enllà de la programació i la robòtica educativa: el pensament computacional en l'ensenyament STEAM a infantil i primària.

Cristina Simarro (cristina.simarro.rodriguez@uab.cat)

Víctor López (victor.lopez@uab.cat)

Centre de Recerca per a l'Educació Científica i Matemàtica (CRECIM). Universitat Autònoma de Barcelona (UAB).

Pere Cornellà (pere.cornellacanals@udg.edu)

Marta Peracaula (marta.peracaula@udg.edu)

Mariona Niell (mniell@silver.udg.edu)

Meritxell Estebanell (meritxell.estebanell@udg.edu)

UdigitalEdu, Universitat de Girona (UdG).

El pensament computacional està guanyant cada vegada més rellevància en el món educatiu. Més enllà de l'interès que pot tenir desenvolupar aquesta forma de pensar i resoldre problemes en un context cada cop més informatitzat, el desenvolupament del pensament computacional pot recolzar l'aprenentatge de totes les disciplines que conformen l'STEAM (Science, Technology, Engineering, Arts and Mathematics). El present article, realitzat en el marc del projecte PECOFIM (Pensament Computacional en la Formació Inicial de Mestres), ofereix un breu resum de les definicions existents a la literatura sobre pensament computacional, proposant uns indicadors que poden ajudar a identificar-lo i promoure'l a l'escola. Alhora, es presenten exemples reals per tal de treballar el pensament computacional a les etapes d'educació infantil i primària, utilitzant-se els indicadors proposats per tal d'analitzar-les.

Paraules clau: Pensament computacional, Bee-bot, Scratch, STEAM

Computational thinking is gaining increasing relevance in education. Beyond the interest that developing this way of thinking and problem-solving problems may have in an increasingly computerized context, the development of computational thinking can support the learning of all STEAM (Science, Technology, Engineering, Arts and Mathematics) disciplines. In the framework of the PECOFIM project (Computational Thinking in Pre-Service Teacher Training), this paper offers a brief summary of the existing definitions in the literature about computational thinking and proposes indicators that can help to identify it and to promote it in school. At the same time, it introduces real examples for dealing with computational thinking in early childhood and primary education, using the proposed indicators to analyse them.

Paraules clau: Computational thinking, Bee-bot, Scratch, STEAM

INTRODUCCIÓ

Robots, impressores 3D, automatismes,... No hi ha dubte que, en els darrers cinc anys, el panora-

ma educatiu, formal i no formal, està patint nombrosos canvis en els quals les tecnologies creatives i la robòtica educativa hi tenen un gran protagonisme. Llenguatges de programació com l'Scratch [1],

kits educatius com el Lego We Do [2] o robots de fàcil programació com els BeeBot [3] són només alguns dels exemples que trobem actualment a l'escola i que evidencien aquesta transformació. El nou currículum de primària del Departament d'Ensenyament (2015) recull també aquesta transformació, incloent la robòtica com a contingut de tecnologia i la programació i la robòtica educativa com a contextos en els quals treballar, per exemple, la resolució de problemes de la competència matemàtica: *“La resolució de problemes és el context ideal per reconèixer les matemàtiques com a eines útils per resoldre situacions, per traduir situacions quotidianes a llenguatge matemàtic, per dissenyar estratègies de resolució, per raonar i justificar les opcions preses, per treballar en grup, arribar a acords i comprendre les raons dels altres i un espai en què els continguts dels diferents blocs s'usen de manera conjunta. Dins d'aquesta dimensió, la programació i la robòtica educativa proveeixen una sèrie d'estratègies que afavoreixen les habilitats esmentades anteriorment”* [4].

Però, quines són aquestes estratègies? Com s'explica el paper central que estan guanyant la programació i la robòtica educativa a les nostres aules? Al nostre parer, la resposta a aquestes preguntes té a veure amb el que alguns anomenen la nova alfabetització del s.XXI: el pensament computacional (Papert, 1980, 1996; Wing, 2006, 2010). Més enllà d'aprendre a programar, el pensament computacional s'explica com una nova forma d'abordar la resolució de problemes que resulta essencial en el moment actual en el qual vivim.

Aquest article ofereix un breu resum de les definicions existents a la literatura sobre pensament computacional, proposant uns indicadors que poden orientar el treball d'aquesta forma de pensar, i també a valorar fins a quin punt l'alumnat la desenvolupa. Alhora, es presenten exemples reals per tal de treballar el pensament computacional a les etapes d'educació infantil i primària, relacionats amb els indicadors proposats. Aquesta publicació és realitzada en el marc del projecte PECOFIM (Pensament Computacional en la Formació Inicial de Mestres), que té com a objectiu que els futurs mestres puguin explorar com el pensament computacional pot ser útil per transformar el treball que es du a terme amb infants.

QUÈ S'ENTÉN PER PENSAMENT COMPUTACIONAL?

L'any 1980 Seymour Papert va introduir el terme “pensament computacional” en el seu llibre

“Mindstorms: Children, computers, and powerful ideas” (traduït a l'espanyol com *“Desafío a la mente”*). El 1996, el mateix Papert, va fer una exposició detallada de la idea en el seu article *“An exploration in the space of mathematics educations”*, però no va ser fins l'any 2006 que Jeannette M. Wing va popularitzar el concepte de pensament computacional en l'àmbit de la investigació educativa i psicològica, publicant un article a la revista *“Communications of the ACM”* (2006). En concret, Wing va argumentar que “el pensament computacional implica la resolució de problemes, el disseny de sistemes i la comprensió de la conducta humana, fent ús dels conceptes fonamentals en la informàtica.” (Wing, 2006: p 33). Anys més tard, Wing (2010) va matisar aquesta definició afirmant que aquesta forma de pensar és aplicable a la resolució de varis problemes, essent una habilitat fonamental per a tota la població i no només per als informàtics i programadors. Des d'aquesta perspectiva, l'autora destaca la necessitat d'integrar les idees computacionals en altres disciplines, plantejant solucions que puguin ésser portades a terme també pels humans (i no només per les màquines).

En els deu anys que han seguit després de la primera publicació de Wing, molts autors han centrat la seva atenció en la idea del pensament computacional aportant definicions complementàries. Brennan & Resnick (2012) proposen entendre el pensament computacional a través de tres dimensions: conceptes computacionals (seqüències, bucles, paral·lelismes, esdeveniments, condicionals, operadors i dades), pràctiques computacionals (incrementar i iterar, testejar i depurar, reutilitzar i combinar, abstroure i modularitzar), i perspectives computacionals (expressar, connectar i qüestionar). Alhora, amb l'objectiu de buscar una definició útil per ser operativa en l'àmbit educatiu, la Internacional Society for Technology in Education (ISTE) i la Computer Science Teachers Association (CSTA) (Stephenson & Barr, 2011), així com Berry (2014), Selby (2013) i Selby & Woollard (2014) han aportat les seves pròpies definicions, que malgrat els matisos, coincideixen en entendre el pensament computacional com la capacitat per identificar problemes que puguin ser resolts d'una manera similar a com ho faria un ordinador, dividir aquests problemes complexos en mòduls de mida inferior, seqüenciar processos llargs i complexos en “passos”, organitzar les dades reconeixent patrons lògics, partir de casos concrets per arribar a situacions abstractes i generalitzables, utilitzar algorismes per automatitzar solucions i avaluar la validesa de les solucions.

De fet, malgrat s'associï el pensament computacional a l'ús de tecnologies digitals, Beauchamp (2016) afirma que aquest ja es fa servir sovint a les escoles, encara que de forma implícita, a l'hora de posar als estudiants davant la resolució de problemes de diferents tipologies (el que s'està popularitzant amb el nom "Aprentatge Basat en Problemes"), i que inclou fases com primer entendre el problema, després plantejar un disseny, pla o projecte, posteriorment executar el pla per resoldre el problema i finalment revisar el resultat. En paral·lel, Resnick (2007) defensa l'estreta relació entre el pensament computacional i el pensament creatiu, associat a com els infants aprenen en l'etapa d'educació infantil, on sovint es dona un procés d'aprenentatge amb un denominador comú: imaginar, crear, jugar, compartir, reflexionar i tornar a imaginar.

MÉS ENLLÀ D'APRENDRE PROGRAMACIÓ I ROBÒTICA

Les diferents propostes que busquen definir el pensament computacional posen especial èmfasi en el paper que aquest juga en el desenvolupament cognitiu i social dels infants. En totes les propostes anteriorment esmentades existeix un consens important en que, més enllà d'aprendre informàtica, el pensament computacional recolza l'aprenentatge d'altres disciplines (ciències, llengües, arts, etc.), ajudant a desenvolupar formes de pensar, fer i relacionar-se, essencials per a aquestes disciplines. Per exemple, la nova definició de "pràctiques científiques i enginyerils" que proposa la National Research Council nord-americana (2012) inclou, com una de les vuit pràctiques STEM (Science, Technology, Engineering and Mathematics) clau, l'ús del pensament computacional. De fet, Weintrop et al (2016) va més enllà, i afirma no només que el pensament computacional és clau en la pràctica STEM, sinó que gràcies a la tecnologia digital aquest està transformant la pròpia naturalesa d'aquestes pràctiques STEM. Per exemple, la biologia computacional, basada en estructures de dades i algorismes, està transformant la pròpia manera de pensar de la biologia com a ciència.

Alhora, Barr & Stephenson (2011) proposen com els conceptes clau del pensament computacional poden aparèixer i incloure's en les activitats educatives de diferents disciplines (matemàtiques, ciències, socials, llengües i arts), i posen com a exemple com la capacitat d'identificar patrons pot ser de gran utilitat en llengües, o com la capacitat de dividir problemes complexos en mòduls de mida inferior i sistematitzar-ho amb l'ús d'algorismes és clau en la resolució de problemes matemàtics. Finalment, Beauchamp et al (2016) posen l'èmfasi en

que el pensament computacional ajuda a desenvolupar algunes de les anomenades competències del s.XXI: creativitat, resolució de problemes i treball en equip.

UNA PROPOSTA D'INDICADORS PER IDENTIFICAR I PROMOURE EL PENSAMENT COMPUTACIONAL A L'ESCOLA

Amb l'objectiu d'oferir al professorat d'educació infantil i primària una proposta operativa que ajudi a identificar i promoure en l'alumnat el pensament computacional entès com una competència, podem resumir les definicions fetes en els darrers anys pels diferents autors esmentats, expressant-les de la següent manera:

- a) Capacitat per identificar i formular problemes que puguin ser resolts utilitzant un ordinador o altres eines.
- b) Capacitat per organitzar i analitzar les dades de manera lògica i reconeixent-hi patrons.
- c) Capacitat per representar dades a través d'abstraccions.
- d) Capacitat per automatitzar solucions a través d'algorismes i passos ordenats.
- e) Capacitat per desglossar i dividir situacions complexes en altres més senzilles.
- f) Capacitat per identificar, analitzar, i aplicar possibles solucions amb l'objectiu d'aconseguir la combinació més eficient i eficaç dels passos i recursos, entenent que hi pot haver diversos camins per arribar a una solució final.
- g) Capacitat per transferir i generalitzar aquest procés de resolució de problemes a una àmplia varietat de problemes diferents.

EXPERIÈNCIES D'AULA

A continuació presentem algunes propostes didàctiques orientades a promoure el pensament computacional amb infants d'infantil i primària, alhora que es treballen diferents continguts curriculars de matemàtiques i coneixement del medi natural, social i cultural integrats amb les arts. Es tracta d'activitats emmarcades en el que es coneix com a STEAM, que inclou les arts a les disciplines STEM, permetent que els estudiants connectin el seu aprenentatge en una dimensió més global, complementant l'enginyeria i la ciència amb les pràctiques artístiques i principis de disseny [5]. Lluny de voler ser exhaustiu, mostrem quatre exemples concrets que, centrant-se en l'ensenyament STEAM, es plantegen com a projectes en què els estudiants

aprenen a prendre riscos, s'involucren en l'aprenentatge experimental, persisteixen en la resolució de problemes, i col·laboren en equips, i naveguen en un procés creatiu. Es poden trobar moltes més propostes implementades en escoles de Catalunya en les jornades "Programa", que anualment organitza el Departament d'Ensenyament conjuntament amb el CityLab de Cornellà [6].

1. Seguir un circuit: Com podem fer que una persona segueixi un camí concret i com ho hem de fer si volem que sigui un Bee-Bot el que es bellugui?

Aquesta és una activitat dirigida a infants de P5 i cicle inicial de primària on es practica la **seqüenciació**, la **lògica** i la **iteració**, i es fa en dos parts: una primera on els infants experimenten amb els seu propi cos, i una segona on traslladen aquestes les vivències a la programació del Bee-Bot. Al llarg del procés, els infants reflexionen sobre les dimensions espacials planes (endavant, endarrere, dreta, esquerra), i fan una descomposició i estructuració mental del moviment (aprenent a dissociar, per exemple, la idea de "gir" i "translació"). El repte que es proposa pot enfocar-se de dues maneres: o bé trobar la trajectòria necessària per anar d'un punt de sortida a un punt d'arribada ja definits, o bé trobar el punt d'arribada, a partir d'un punt de sortida i un seguit de passos ja definits.

La primera part de l'activitat consisteix en jugar amb els infants a fer de robots i programadors, per torns. Els que fan de programadors hauran de donar instruccions precises per fer bellugar als que fan de robots per un espai concret, de la manera que els segons hauran de seguir les instruccions sense equivocar-se. Per jugar-hi, només necessitem una quadrícula prou gran, que la podem representar en qualsevol espai: la pròpia classe, el gimnàs o el pati, per exemple. En el cas d'una aula, podem establir que la quadrícula la formin les pròpies rajoles, i si fem l'activitat al pati, podem dibuixar la quadrícula a terra amb un pal o amb guix. Entre tots hauran de trobar aquelles instruccions que els són més efectives: avançar, recular, girar a dreta i esquerra, aturar-se, etc. A més, si es vol afegir complexitat a l'experiència, podem afegir obstacles que no es poden traspasar o objectes que cal recollir. Un dels avantatges de començar "fent de robots" amb el propi cos és que l'activitat pot resultar més vivencial per a l'infant, tal com proposa la filosofia "unplugged programming", és a dir, de programació sense endolls.



Figura 1. Detall de la Bee-bot i la catifa per la qual pot desplaçar-se.

La segona part de l'activitat consisteix en aplicar tots els aprenentatges realitzats per programar el Bee-Bot, un petit robot en forma d'abella, per a què es mogui per una catifa marcada per una graella de quadres sortint d'un punt i arribant a un objectiu concret. Els moviments de la Bee-Bots sempre són iguals. Aquest disposa de 4 tecles de moviment: endavant (15 cm), endarrere (15 cm), gir a la dreta i gir a l'esquerra (amb les quals el robot gira 90° respecte el seu centre sense traslladar-se). També hi ha una tecla de pausa, que atura el robot durant 1 segon, una altra que esborra el que tingui memoritzat (clear) i una que serveix per executar les ordres donades (go). Per fer-lo funcionar, l'infant pensa l'itinerari que ha de fer el Bee-Bot per arribar a l'objectiu des del punt d'inici, i seguidament prem la seqüència de tecles necessàries per fer tot el camí (el robot pot recordar fins a 40 instruccions). Amb la tecla "go" s'engega el robot, i es pot comprovar com el programa introduït aconsegueix l'objectiu proposat. Les catifes es poden dissenyar de manera que representin escenaris on el robot viurà la seva història: un paisatge urbà (on el robot fa diversos encàrrecs seguint un itinerari), un terreny muntanyós (on el robot ha d'anar a buscar un tresor a una cova evitant els obstacles: riu, camins sense sortida,...), una graella de lletres (amb les que el robot ha d'escriure un text: el nom d'algun dels infants, paraules relacionades amb conceptes treballats,...), una graella de números (per practicar operacions: sumes, restes,...). Algunes d'aquestes catifes es comercialitzen però també se'n poden crear de pròpies modificant catifes en blanc (pintant-hi, posant-hi plantilles, objectes, obstacles, fent-hi un cau,...). Les catifes i els reptes els pot dissenyar el professorat per adaptar-los als continguts que vulgui treballar, o també implicant al grup classe per crear propostes col·lectives. Finalment, el Bee-Bot també es pot disfressar per crear altres personatges.



Figura 2. Seqüència d'instruccions per definir el moviment del robot. El kit de Bee-Bot porta fitxes de cartró per representar cadascuna de les instruccions, però també es poden crear a classe o escriure les instruccions amb llapis i paper.

2. La geometria dels polígons: Quins polígons podem trobar en els objectes que ens envolten i com els expressem matemàticament?

Aquesta experiència està dirigida a infants del cycle superior d'educació primària i té com a objectiu que els alumnes reconeguin figures geomètriques planes regulars en el seu entorn real i descriu matemàticament el seu contorn amb el llenguatge de programació Scratch (Resnick, 2009).

En una primera part, es demana als alumnes que es passegin pel centre o pels seus voltants, amb l'objectiu d'observar, descobrir i realitzar fotografies de figures geomètriques planes amb un dispositiu digital mòbil proporcionat a cada grup. Un cop a l'aula es formen equips de treball, repartits entre els ordinadors o tauletes que hi hagi disponibles, i se'ls demana que transfereixin a l'ordinador les fotografies realitzades en les quals hi apareguin objectes amb formes geomètriques.

A partir d'aquest moment els alumnes comencen a treballar amb Scratch. Si no tenen coneixements previs d'aquest llenguatge de programació necessiten familiaritzar-se amb unes quantes eines que es poden conèixer mitjançant una activitat de descoberta d'uns 20 minuts.



Figura 3. Estudiant fotografiant una rotonda.

Cada grup tria una de les fotografies realitzades anteriorment per fer-la servir de fons del seu programa. La poden editar amb el mateix Scratch per a que la figura geomètrica estigui centrada i poden afegir-hi les seves creacions artístiques. A continuació trien un personatge de la galeria d' Scratch, que serà el que es programarà per a que es mogui per la pantalla tot traçant el contorn del polígon. Durant la programació del personatge, els alumnes necessiten aplicar elements de pensament computacional, com **esdeveniments** i **bucles**. Fan servir el mètode heurístic, per exemple, per trobar la llargada del recorregut de cada costat i l'angle de gir per resseguir correctament el contorn del polígon.

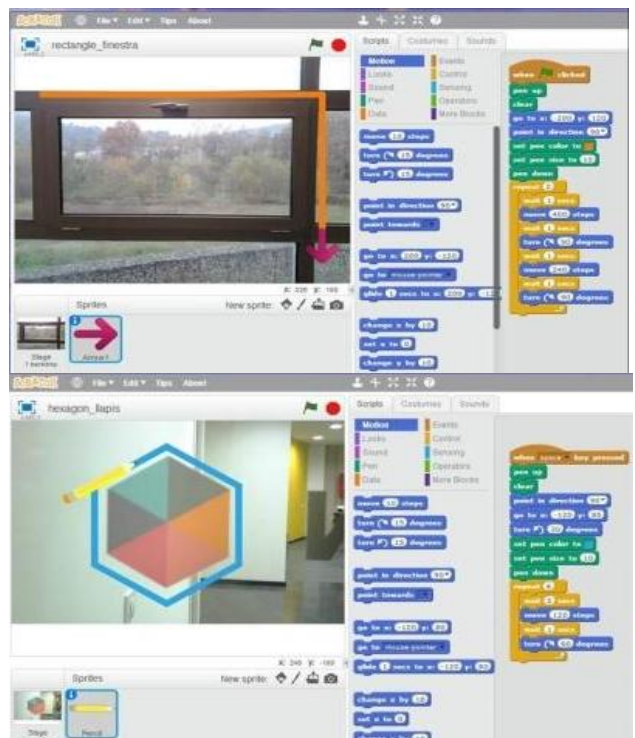


Figura 4. Dos exemples de expressions de formes geomètriques a través de Scratch.

En les nostres experiències els infants han programat el contorn de quadrats, rectangles, triangles i hexàgons. Alguns alumnes han verbalitzat que l'angle que han d'introduir en el programa és l'angle exterior, moment en que s'ha incentivat la reflexió sobre angles exteriors i interiors. També s'ha reflexionat i provat conjuntament que com més costats afegim a un polígon regular més ens aproximem a la figura d'un cercle. Paral·lelament a l'activitat de programació es pot oferir als nens un espai amb escuradents i plastilina on poder crear, manualment, les seves formes geomètriques.



Figura 5. Estudiants treballant amb escuradents i plastilina per definir les formes geomètriques.

3. La rotació del Sol: Com podem saber quan triga el Sol en girar sobre ell mateix fent una pel·lícula de les taques solars?

Aquesta experiència es pot realitzar amb alumnes de cicle superior d'educació primària o primers cursos d'educació secundària. En ella els alumnes reproduïen un descobriment científic històric sense saber-ne el resultat previ. En concret, es tracta de muntar una pel·lícula amb Scratch amb imatges del Sol preses en dies consecutius pel satèl·lit de la NASA Solar Dynamics Observatory per descobrir quan triga en girar sobre sí mateix. Al muntar la pel·lícula observen que les taques del Sol es van movent de posició aparent i, a partir d'aquí, poden deduir que l'astre gira sobre ell mateix i poden trobar el període de rotació.

Es comença mostrant als alumnes els dibuixos i esbossos que va fer Galileu del Sol, sense esmentar de quin astre es tracta. Ells mateixos dedueixen que és el Sol i, al igual que Galileu va fer, dedueixen que està en rotació, ja que la posició aparent de les taques solars que va dibuixar va canviant

amb els dies. Amb una bola de porexpan groga que representi el Sol i una xinxeta fosca clavada a la bola per representar una taca es pot escenificar el canvi de posició aparent de la taca a mida que el Sol va rotant. A més, si hi ha la possibilitat, és molt aconsellable que els alumnes, durant l'activitat, puguin anar a l'exterior a observar el Sol amb l'aparell SolarScope, un petit telescopi amb el qual es pot veure el reflex del Sol i les seves taques sense perill per a la vista.



Figura 6. Projecció de la ombra solar a la superfície d'un SolarScope.

Un cop feta aquesta introducció, i un petit recordatori i intercanvi de coneixements previs, els alumnes formen equips de 2 o 3 integrants per distribuir-se als ordinadors. En aquest moment se'ls pot ensenyar a accedir a la base de dades d'imatges reals del Sol del satèl·lit de la NASA Solar Dynamics Observatory. Les poden triar per rangs de dates (és important que n'hi hagi al menys una per dia durant un mes seguit). És interessant que cada grup es baixi rangs de dates diferents per poder comparar resultats. També es poden tenir les imatges prèviament baixades a l'ordinador. Finalment els alumnes poden muntar la pel·lícula amb l' Scratch.

Amb la pel·lícula muntada, i coneixent la data de cada imatge, poden deduir el període de rotació del Sol veient el dia que una taca concreta surt per un costat del Sol, i com al cap d'uns dies s'amaga per l'altre. En aquest moment acaben de fer el mateix descobriment que va fer Galileu amb els seus dibuixos fa 400 anys: descobreixen que el Sol gira sobre ell mateix cada 24 dies aproximadament.

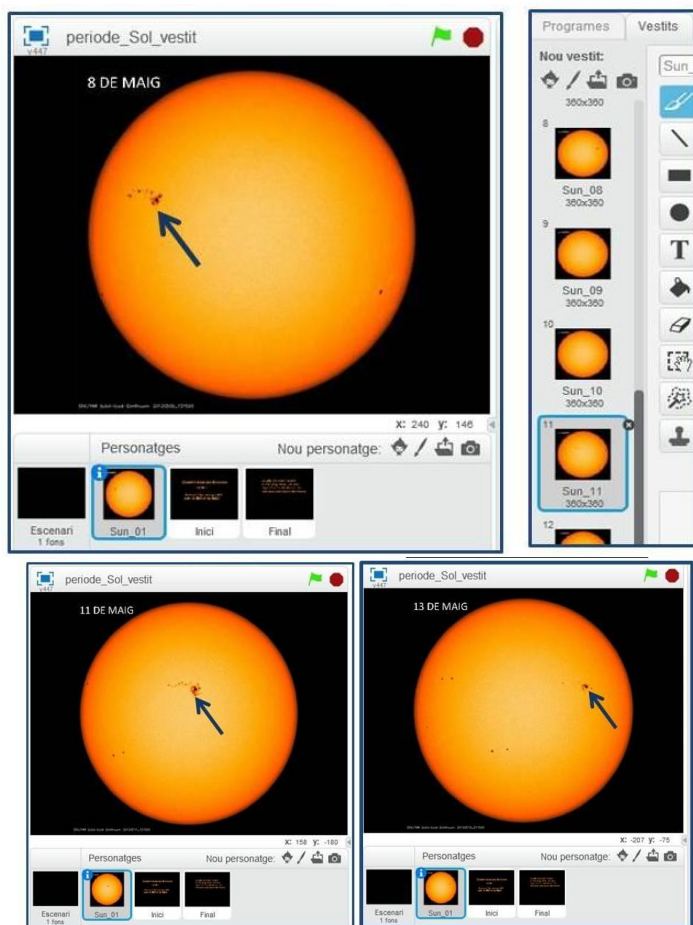


Figura 7. Diferents imatges d’una mateixa taca solar en diferents dates, que permeten calcular el període de rotació del Sol.

En aquesta experiència els alumnes desenvolupen la pràctica científica i reflexionen sobre com l’observació metòdica i sistemàtica dels fenòmens de la naturalesa ens ajuda a entendre el món on vivim. Al muntar la pel·lícula amb un programa fet per ells amb Scratch, a més, necessiten utilitzar esdeveniments per sincronitzar la capçalera, l’inici de la pel·lícula i el final i necessiten **utilitzar seqüències dins d’un bucle** per deixar el mateix interval de temps entre fotogrames.

4. L’alimentació del gat: expressar un model científic escolar sobre nutrició a través d’un model computacional en forma de joc.

Aquesta experiència duta a terme per una mestra en formació consisteix en el disseny d’un petit videojoc per infants de 1r cicle de primària, tot i que aquest disseny es podria dur a terme amb estudiants de cicle superior de primària que elaboressin

un videojoc semblant pels estudiants dels primers cursos de primària. Aquest petit videojoc dissenyat amb Scratch tracta sobre alimentació saludable, i permet observar la resposta del gat de Scratch (el personatge per excel·lència d’aquest llenguatge de programació) en funció de quins aliments consumeix.

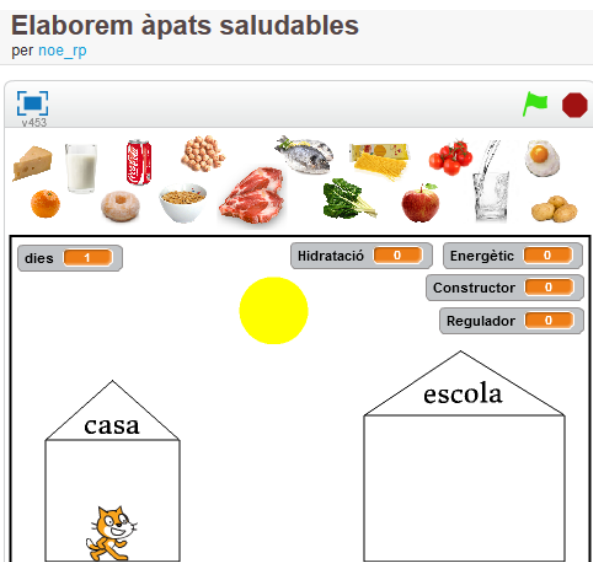


Figura 8. Aspecte del joc “Elaborem àpats saludables” per a infants de 2n curs de primària.

Per elaborar les regles del joc, era necessari definir un model científic escolar i expressar-lo mitjançant **variables** relacionades entre elles mitjançant **algorismes**. En el cas que presentem, la mestra va centrar-se en les principals funcions nutricionals dels aliments (funció estructural / constructora, funció energètica i funció reguladora, sumada a la funció hidratació) i les va convertir en variables amb un valor numèric del zero al cinc. El programa dissenyat modificava els valors numèrics de cada una de les variables en funció de l’alimentació del gat al llarg dels dies.

Posteriorment, fent servir funcions de **condicions amb operadors matemàtics**, va definir la reacció del gat davant dels diferents tipus d’alimentació, la qual s’expressava visualment a través dels diferents vestits que permet el llenguatge Scratch, tal com s’exposa en la Figura 9. Així, en funció del tipus d’aliment, el gat podia estar cansat (si havia ingerit pocs aliments que tenen funció energètica), pesat (si n’havia ingerit massa), restret (si havia ingerit pocs aliments que tenen funció regulació), deshidratat, etc.








Quan passa?	Vestits	Què diu?	Què fa?
Normal		Pregunta pels diferents àpats	Camina ràpid
Energètic = 2		"Tinc tanta energia que podria córrer una cursa de 100 metres!"	Es mou i camina ràpid
Constructor > 2		"He crescut i em noto fort!"	Creix i camina ràpid
Regulador = 4		"Estic súper bé! lupill!"	Salta i camina ràpid
Energètic < 2		"Estic molt cansat"	Camina lentament
Energètic > 2		"Em noto pesat, crec que m'he engreixat"	Camina lentament
Constructor < 2		"Aii! M'he lesionat"	Camina lentament
Regulador < 4		"Em fa molt de mal la panxa. Vaig al lavabo i no faig res.."	X
Hidratació < 2		"He de beure més aigua per hidratar-me!"	X

Figura 9. Expressió del model de nutrició a través de variables i relacions entre variables.

REFLEXIONS FINALS

Tal com hem pogut veure, les activitats presentades evidencien l'ampli rang d'activitats que podem proposar a l'aula per treballar el pensament computacional. Totes elles utilitzen suports diversos i, si bé les quatre activitats plantejades es basen en l'ús un suport digital, en l'activitat 1 i la 2 les activitats impliquen una part important de treball "unplugged", ja sigui a través del moviment corporal o de material manipulatiu. Alhora, comprovem que podem proposar activitats per a edats molt diverses, des d'infantil fins a cicle superior. A més, les activitats es poden plantejar de manera que s'abordin diverses disciplines (matemàtiques, ciències, arts,...) i es treballin algunes de les competències clau per al s.XXI com el treball en equip o la creativitat.

Centrant-nos en el desenvolupament del pensament computacional, podem analitzar les diferents experiències presentades fent servir els indicadors del pensament computacional anteriorment exposats, identificant quin dels seus aspectes es promou en cada cas. Així, en l'activitat 1 està especialment dissenyada per promoure la capacitat

per identificar i **formular un problema** que pugui ser resolt fent servir una eina programable (en aquest cas la Bee-bot), i també per treballar la capacitat per seqüenciar les instruccions en forma de passos ordenats. Alhora, es treballa la idea que hi pot haver **diversos camins per arribar a una solució final**. L'activitat 2 promou la capacitat per automatitzar els processos d'elaboració d'una forma geomètrica fent servir conceptes computacionals com ara **esdeveniments** i **bucles**, però també promou la modularització o desglossament del problema complex en passos més senzills. En l'activitat 3 es promou la capacitat per organitzar i analitzar les dades de manera lògica i reconeixent-hi **patrons**, així com el **procés iteratiu de refinament** de la solució, per exemple a l'hora d'ajustar els fotogrames. Finalment, l'activitat 4 promou la capacitat per representar dades **a través d'abstraccions**, com són les variables relacionades amb les funcions nutricionals i la seva relació a través d'operadors matemàtics ">", "<" i "=", i l'ús d'**algoritmes per automatitzar** el comportament del gat davant de cada situació nutricional.

Amb aquest primer article del projecte PECO-FIM (Pensament Computacional en la Formació Inicial de Mestres) hem volgut oferir al professorat d'infantil i primària interessat en aquest enfocament educatiu una visió àmplia sobre el pensament computacional, incidint no només en la seva definició sinó en els perquè de la seva rellevància. Alhora, s'ha proposat una definició operativa que pugui ser útil a l'hora d'analitzar propostes educatives des de la perspectiva del pensament computacional. Finalment, s'ha pretès exemplificar aquests conceptes a través d'eines i exemples útils per la pràctica docent, buscant animar al professorat a identificar què del que ja està fent pot incidir en el pensament computacional a l'aula i inspirar-lo en el disseny de propostes educatives adreçades a desenvolupar entre el seu alumnat alguns dels aspectes clau d'aquesta forma de pensar i resoldre problemes.

BIBLIOGRAFIA

- BARR, V., STEPHENSON, C. (2011). *Bringing Computational Thinking to K-12: What is Involved and What is the Role of the Computer Science Education Community?* ACM Inroads, 2(1), 48–54.
- BEAUCHAMP, G. (2016). *ICT and computing in the primary school*. Computing and ICT in the Primary School: From pedagogy to practice (p. 252).

- BERRY, M. (2014). Does not compute. Computational Thinking in Primary Schools. *Teach Primary*.
- BRENNAN, K., RESNICK, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association* (pp. 1–25). Vancouver. Retrieved from http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf
- PAPERT, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. The Harvester Press.
- PAPERT, S. (1996). An Exploration in the Space of Mathematics Educations, *International Journal of Computers for Mathematical Learning*, Vol. 1, No. 1, pp. 95-123.
- SELBY, C. C. (2013). *Computational Thinking: The Developing Definition*. ITiCSE Conference 2013.
- SELBY, C. C., WOOLLARD, J. (2014). Refining an Understanding of Computational Thinking.
- STEPHENSON, C., BARR, V. (2011). Defining Computational Thinking for K-12. *CSTA: The Voice of K-12 Computer Science and Its Educators*, 7(2).
- WEINTROP, D., BEHESHTI, E., HORN, M., ORTON, K., JONA, K., TROUILLE, L., WILENSKY, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. *Journal of Science Education and Technology*, 25(1), 127–147. <http://doi.org/10.1007/s10956-015-9581-5>
- WING, J. M. (2006). Computational Thinking. *Communications of the ACM*, 49(3), 33–35. <http://doi.org/10.1145/1118178.1118215>
- WING, J. M. (2010). Computational Thinking: What and Why? Retrieved from <http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>

NOTES

- [1] <https://scratch.mit.edu/>
- [2] <https://education.lego.com>
- [3] <https://www.bee-bot.us/>
- [4] <http://www.mecd.gob.es/dms/mecd/educacion-mecd/mc/lomce/mapa-ccaa/ordenes-primaria/primaria-catalunya.pdf>
- [5] <http://educationcloset.com/steam/what-is-steam/>
- [6] <http://blocs.xtec.cat/jornadaprograma/>