BATXILLERAT 2n curs. Programació i tractament de dades II

Vols ser un “bon hacker”?

Competències específiques i criteris d’avaluació vinculats

Competència específica 1 a desenvolupar

1. Desenvolupar algoritmes i aplicacions informàtiques en diferents entorns, aplicant els principis del pensament computacional i incorporant les tecnologies emergents, per crear solucions a problemes concrets, independentment del llenguatge utilitzat.

Aquesta competència específica es connecta amb els següents descriptors del Perfil de sortida: STEM1, STEM2, STEM3, CD3, CD4, CD5, CPSAA1.1, CPSAA5.

Criteri d’avaluació

1.3. Calcular el cost computacional d’un programa en funció de les dades d’entrada i, a partir de l’anàlisi dels resultats obtinguts, realitzar les modificacions necessàries per millorar-ne l’eficiència.

Competència específica 2 a desenvolupar

5. Publicar i documentar en diferents formats els programes desenvolupats i les dades generades de forma clara i precisa per poder ser emprades pels usuaris.

Aquesta competència específica es connecta amb els següents descriptors del Perfil de sortida: CCL1, STEM1,STEM2, STEM4, STEM5, CD2, CD3, CD4, CD5, CPSAA1.1, CPSAA1.2, CPSAA5.

Criteri d’avaluació

5.2. Estudiar si els programes creats resulten útils i usables per a la seva audiència objectiva, identificant les mancances i, segons el cas, aplicant millores d’utilitat i d’usabilitat.

Competència específica 3 a desenvolupar

6. Realitzar simulacions d’entorns reals per tal de conèixer les diferents branques de la informàtica.

Aquesta competència específica es connecta amb els següents descriptors del Perfil de sortida: STEM1, STEM2, STEM3, STEM4, STEM5, CD2, CD3, CD4, CD5, CPSAA1.1, CPSAA5, CE3.

Criteri d’avaluació

6.1. Incorporar conceptes de certes branques de la informàtica en la realització de programes i aplicacions.

6.4 Extreure conclusions objectives a partir d’evidències detectades en l’execució dels programes informàtics.

Sabers bàsics

– Algorísmia. Descomposició del problema. Disseny.

- Cost computacional. Iniciació i estructures de control eficients.
- Selecció del tipus de dades en funció del context plantejat.
- Utilitat. Eficiència.
- Presentació de la informació.

Context o contextos

Professional i escolar.

Dinàmiques de presa de consciència

En cursos anteriors, sobretot a la matèria de Recursos digitals, l’alumnat haurà pres consciència sobre la importància de les seves dades, identificant les principals amenaces per tal de fer un ús responsable, saludable i legal.
La seguretat informàtica és cada vegada més present a les nostres vides.
La primera barrera per protegir-nos de l’accés indesitjat és l’ús de contrasenyes. En aquest punt, és important conèixer quins aspectes faran que una contrasenya sigui robusta. Així, s’han d’evitar les contrasenyes més emprades[1] i les que són més fàcils de hackejar.
Ara és l’hora d’analitzar com es construeixen les contrasenyes segures i veure com podem explicar a la resta de ciutadans si una contrasenya és segura o no.
Una primera activitat en grup podria ser identificar quines són les contrasenyes més usades, creant un diccionari de contrasenyes i

1https://es.statista.com/grafico/23636/contrasenas-mas-usadas-en-el-mundo/

cercar patrons per intentar hackejar-les.

Instruccions i material

Després d'informar-se de les característiques que ha de tenir una bona contrasenya, els alumnes han de plantejar una estratègia de comprovació de contrasenyes i realitzar un programa que analitzi la robustesa d'una contrasenya donada.
A més, han de calcular el cost computacional de trobar la contrasenya en funció de la seva complexitat.
També han d'elaborar una web per mostrar de forma gràfica i senzilla els punts que ha de complir una contrasenya robusta i les pautes que s'han de tenir en compte per a la seva creació. De manera opcional, es pot oferir a l'usuari la possibilitat de comprovar la robustesa d'una contrasenya introduïda.
La codificació es durà a terme amb el llenguatge de programació consensuat entre el professor i els alumnes juntament amb l'entorn de desenvolupament integrat (IDE).

Descripció i planificació de la tasca o tasques. Tipus i característiques del producte o productes obtinguts

Crear un algoritme de comprovació de contrasenyes. Els alumnes, amb l'orientació del professor, estableixen fins a quin nivell de complexitat s'avaluarà la contrasenya, si s'ha de combinar una cerca en un diccionari o llista de contrasenyes més utilitzades amb força bruta o el nombre de comprovacions que s'haurien de fer.
En funció de les dades d'entrada, calcular el nombre d'intents són necessaris per trobar una contrasenya similar[2].
Investigar què suposaria l'adopció de la verificació en dues passes mitjançant l'enviament d'un missatge al correu electrònic o al telèfon mòbil.
Dissenyar una web o aplicació per calcular la robustesa d'una contrasenya, tenint present l'usuari final a qui va destinada, de forma que la informació sigui útil i precisa.

2Per exemple, si només són dígits: 10^x on x és el nombre de dígits de la contrasenya

Procediments de retroacció, revisió i supervisió per part del professorat durant la realització de la tasca o tasques

En el debat, els alumnes reben de forma immediata retroacció a les propostes i determinen l'algoritme a seguir per implementar el programa. Juntament amb el professor, s'encaminen les propostes perquè s'adeqüin a l'objectiu cercat. Professor i alumnes defineixen les bases d'orientació que determinaran el que s'ha de fer per dur a terme les tasques.
Mitjançant el control de versions el professor pot revisar el codi i donar una retroacció als alumnes.

Procediment d'avaluació final, qualificació i transferència del coneixement

A partir de la graella d'observacions, es definiran graelles d'autoavaluació perquè els alumnes es puguin autoavaluar.
Els alumnes podran comparar les diferents estratègies dels grups per veure quin algorisme troba amb menys intents un conjunt de contrasenyes.
L'aplicació web generada podrà ser d'utilitat a la matèria de Recursos Digitals d'ESO per veure la robustesa d'una contrasenya.