

Sprint 3 → Tasca 2

Crea un algoritme. Inscripció a l'escola

Enunciat

L'escola d'educació primària del nostre barri obren ara el termini d'inscripció. Tenen moltes dades a registrar per a cada nen que s'inscriu i volen automatitzar la recollida de dades per tal de que no els porti tanta feina, evitar errades i fer possible en un futur que es pugui consultar la puntuació per Internet.

Cada nen que s'inscriu té un número de punts segons alguns criteris. Els criteris per al càlcul dels punts són els següents:

1. Si té almenys un germà al centre, té 40 punts.
2. Per proximitat:
 - Si viu a la mateixa població, té 30 punts
 - Si el pare o mare treballa a la mateixa població, té 20 punts
(en cap cas es sumen aquests dos criteris, es tria el que doni més puntuació)
3. Si té alguna discapacitat igual o superior al 33% o malaltia crònica: 10 punts
4. Si forma part d'una família nombrosa o monoparental: 15 punts
5. Si el pare, mare, tutor legal o germà ha estat escolaritzat a la mateixa escola: 5 punt.

Per exemple, un nen de família nombrosa que viu al mateix barri de l'escola tindria 30(proximitat) + 15 punts (per família nombrosa) = 45 punts.

I un nen d'una altre població que té ja un germà a l'escola tindrà 40 punts.

Cal que definiu l'algoritme del programa que demanarà les dades a l'usuari. les analitzarà i en funció d'això determina els punts del nen i els mostra per pantalla.

Per a cada nen, per a preservar la confidencialitat, inicialment entrarem un número, que correspon al número d'ordre de la seva inscripció. Després haurem de fer les preguntes per demanar les dades que ens serviran per a determinar els punts que ha de tenir.

Recomanació: per a qualsevol pregunta que heu de fer, per exemple: Té algun germà al centre? Oferiu dos possibles respostes: 0 per no i 1 per a si. Un exemple de funcionament del programa seria aquest:

```
Num. de registre del nen: 25
Té algun germà al centre (0: no / 1:si)? : 0
Viu a la zona de l'escola (0: no / 1:si)? : 1
...
```

Feu l'algoritme en llenguatge natural, tal i com ho teniu als exemples. Frases curtes i que cada frase representi una acció. Si trobeu que en una frase es descriuen diverses accions plantegeu-vos si cal separar-la en diferents passes. Teniu un exemple al final d'aquest enunciat.

Lliurament

Què

Heu de :

1. Repositori de github.
2. Arxiu amb l'algoritme que heu creat.

Com

El nom del fitxer de l'algoritme serà: Cognom1Inicial del cognom2. Els cognoms s'escriuran sense accents. Per exemple, l'estudiant *Joan García Santos* posaria el següent nom al seu fitxer de l'exercici: GarciaS.doc o bé GarciaS.pdf.

Autoavaluació

Avalua el teu programa segons els [criteris](#) de la taula d'avaluació i omple aquí quina ha estat la teva puntuació

Granularitat:

Claredat i ordre:

Indentacions:

Correcte:

Puntuació final:

Criteris per avaluar l'algoritme

Criteri	2 Notable	1 Suficient	0 Insuficient	puntuació
Granularitat	La majoria de les passes que s'indiquen reflecteixen una única acció	Una part de les passes fan referència a una única acció, però també hi ha d'altres que són complexes (contenen més accions en una sola frase)	La majoria de les passes que s'indiquen fan referència a accions complexes (que es podrien descomposar en accions més petites)	
Claredat i ordre	Es pot seguir fàcilment. És endreçat i s'entén la lògica de l'algoritme. Hi ha comentaris indicant que es fa a cada part.	A trossos s'entén i altres trossos no són gaire clars, es complica No hi ha comentaris	És complicat i desendreçat.	
Indentacions	L'algoritme està correctament indentat.	En general està indentat però hi ha algunes errades d'indentació.	No està gens indentat.	
Correcte	Fa la majoria de coses que es demanen	Fa una part de les coses que es demanen, però no arriba a la majoria	Fa alguna cosa de les que es demanen, però la majoria no	
Correcció en el lliurament	El lliurament segueix les pautes donades	No copia l'informe a l'espai text en línia.	No lliura res del que es demana.	

Exemple

Partim del següent cas:

Volem fer l'algoritme per resoldre el pagament d'una entrada de cinema.

El preu general és de 8,5€. Però si té carnet de fidelitat del cinema, qualsevol dia es de 4,9€.

Quan sabem quan ha de pagar, hem de mirar si els diners que té el client són prou per a fer el pagament. Si té més, hem de calcular quan cal tornar. Si no et té prou, l'hem d'avisar de quants més diners ens ha de donar.

Per fer l'algoritme, ordenaríem lògicament les condicions (si es pot aplicar la tarifa que resulta més econòmica, primer s'aplica aquesta) i després aniríem detallant pas a pas cada acció que hem de fer.

L'algoritme resultant seria el següent:

```
//Definim tipus de dades
sencer carnetFidelitat
real dinersQueTeClient, preu, dinersQueFalten, dinersATornar

//Demanem dades del client
mostrar: Entra diners que tens:
dinersQueTeClient = llegir real
mostrar: Té carnet fidelitat (0:no/1:si)?
carnetFidelitat = llegir sencer

//Calculem i mostrem
si te carnetFidelitat
    preu ← 4,9
si no
    preu ← 8,5€
si preu < dinersQueTeClient
    dinersATornar ← dinersQueTeClient - preu
    mostrar a l'usuari: dinersATornar
si no
    dinersQueFalten ← preu - dinersQueTeClient
    mostrar a l'usuari: dinersQueFalten
```

Fixeu-vos en algunes coses:

- És important que a cada línia hi hagi només una acció

si te carnetFidelitat preu ← 4,9
Seria INCORRECTE

- Si volem mostrar un càlcul , és millor primer fer-lo i després mostrar el resultat del que s'ha calcular.

mostrar a l'usuari: dinersATornar ← dinersQueTeClient - preu
Seria INCORRECTE

- Tot el que es fa quan es compleix una condició va «dintre» d'aquella condició:

```
si preu > dinersQueTeClient
    diners ATornar ← dinersQueTeClient - preu
    mostrar a l'usuari: dinersATornar
```

Tot el que hi ha al requadre només s'executa quan es compleix la condició de dalt (si preu > dinersQueTeClient), per això està indentat (amb un sagnat més gran) i justament a sota de la condició.

Aquest algoritme és només una possible solució, n'hi poden haver altres correctes.

L'enunciat és intencionalment simple (per exemple, no té en compte si el client té exactament la mateixa quantitat de diners que el cost de l'entrada). Això és per simplificar l'algoritme final.