

Fonaments de Programació (104337)

Curs 2019-2020 - Examen Parcial (29 d'octubre de 2019)

Nom estudiant:

NIU:

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, els procediments i funcions han d'estar ben programats (utilitzant les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Els exercicis 1 a 7 fan referència a un mateix tema que introduïm a continuació:

Un hospital materno-infantil ens demana fer un programa per a gestionar algunes dades mèdiques dels nadons nascuts a l'hospital. Més concretament, es vol guardar i gestionar la informació referent a l'alçada dels nadons al néixer.

El rang d'alçades a considerar és el que va des de 31 cm fins 60 cm. En cas que el nadó mesuri menys de 31 cm o més de 60 cm l'alçada s'introduirà com a 31 o 60 respectivament. Les anotacions sempre es fan en centímetres (cm) i es guardaran com a valors enters.

Cada nadó està identificat amb un codi numèric de 5 xifres. La informació que es vol guardar de cada nadó és el seu codi i la seva alçada.

A més, a cada nadó se li assigna un grup (A,B,C,D) en funció de l'alçada que té. Aquest grup, també s'ha de guardar per a cada nadó.

Totes aquestes dades es guardaran en tres llistes (una pels identificadors dels nadons, una altra per les alçades i una altra pel grup al que pertany a cada nadó). L'ordre en el que es guarden les dades és el mateix en les tres llistes, és a dir, l'identificador d'un nadó, la seva alçada i el grup al que pertany, estan en el mateix índex de la llista corresponent.

Exercici 1 (1 punt)

Fer una funció anomenada `buscar_element` que rebi com a paràmetre una llista d'enters i un element a buscar, i comprovi si l'element es troba a la llista. Si l'element és a la llista, la funció retornarà l'índex de la posició on s'ha trobat l'element. Si no hi és, retornarà -1.

Important: NO ES PODEN UTILITZAR FUNCIONS DE MANIPULACIÓ DE LLISTES, EXCEPTE LA FUNCIO `len()`

```
def buscar_element(llista,element):
    i=0;
    trobat=False
    while((i<len(llista))and(not trobat)):
        if llista[i]==element:
            trobat=True
        else:
            i+=1
    if trobat:
        return i
    else:
        return -1
```

Exercici 2 (1 punt)

Fer una funció anomenada `calcular_grup` que calculi el grup (A,B,C,D) que se li ha d'assignar a un nadó en funció de la seva alçada. La funció rebrà l'alçada en centímetres d'un nadó com a paràmetre i retornarà el caràcter corresponent al grup que correspon a aquella alçada segons el següent criteri:

- 'A' si l'alçada del nadó és inferior a 35cm
- 'B' si l'alçada del nadó es troba en l'interval [35cm, 40cm)
- 'C' si l'alçada del nadó es troba en l'interval [40cm, 50cm)
- 'D' si l'alçada del nadó és igual o superior a 50cm

Nota: Considereu que l'alçada mai serà negativa (no cal fer la comprovació).

```
def calcular_grup(alçada):  
    if (alçada<35):  
        return 'A'  
    elif (alçada<40):  
        return 'B'  
    elif (alçada<50):  
        return 'C'  
    else:  
        return 'D'
```

Exercici 3 (1 punt)

Fer un procediment anomenat `gestionar_dades` que permeti introduir les dades d'un nadó nou. El procediment rebrà com a paràmetres les llistes amb els identificadors dels nadons, la llista de les alçades dels nadons i la llista del grup al que pertany cada nadó.

El procediment demanarà l'identificador numèric d'un nadó i comprovarà si es troba a la llista d'identificadors (amb la funció `buscar_element` de l'exercici 1).

- Si no es troba l'identificador a la llista, s'afegirà l'identificador a la llista d'identificadors i es demanarà l'alçada del nadó. Si el valor introduït és menor o igual que zero, es mostrarà un missatge d'error ("ERROR: Valor incorrecte per l'alçada") i es tornarà a demanar d'introduir l'alçada del nadó. Aquest procés es repetirà fins que s'introdueixi un valor correcte.

Un cop llegida una alçada correcta, es calcularà el grup d'alçada (A, B, C, o D) que se li ha d'assignar (amb la funció `calcular_grup` de l'exercici 2) i s'afegiran les dades a les llistes corresponents.

- Si l'identificador introduït ja es troba a la llista d'identificadors, s'informarà a l'usuari de que ja hi ha les dades d'aquest nadó, i les mostrarà per pantalla (alçada i grup).

```
def gestionar_dades(identificadors,alçades,grups):
    ident=int(input("Introdueix l'identificador del nadó: "))
    i=buscar_element(identificadors,ident)
    if i==-1:
        identificadors.append(ident)
        alçada=int(input("Introdueix l'alçada del nadó: "))
        while (alçada<=0):
            print("ERROR: Valor incorrecte de l'alçada")
            alçada=int(input("Introdueix l'alçada del nadó: "))
        grup=calcular_grup(alçada)
        alçades.append(alçada)
        grups.append(grup)
    else:
        print("El nadó ja està al sistema. Les seves dades són: ")
        print("Alçada: ",alçades[i])
        print("Grup: ",grups[i])
```

Exercici 4 (1 punt)

Fer una funció anomenada `histograma` que rebi la llista de les alçades dels nadons i faci el recompte de quants nadons tenen cadascuna de les alçades entre 31 i 60. Aquest recompte es retornarà en una llista que contindrà el número de nadons amb alçada 31 a la primera posició, el número de nadons amb alçada 32 a la segona, i així successivament.

La llista on es guardarà el recompte, s'ha de crear al principi de la funció amb zeros en totes les posicions.

```
def histograma(alçades):
    histo=[0]*30
    for x in alçades:
        histo[x-31]+=1
    return histo
```

Exercici 5 (1 punt)

Fer una funció anomenada `estadistica` que permeti obtenir algunes mesures estadístiques sobre les dades guardades en una llista. La funció rebrà una llista amb valors numèrics, i calcularà la mitjana i la variància de les dades.

Per a calcular la mitjana, suposeu que disposem d'una FUNCIO JA IMPLEMENTADA:

`mitjana(llista) →` calcula i retorna la mitjana de la llista passada com a paràmetre

Recordeu que per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:

$$variància = \frac{1}{N-1} \sum_i (x_i - \mu)^2$$

on x_i són els elements del conjunt de dades, μ és la mitjana de les dades i N és el nombre d'elements del conjunt de dades.

La funció retornarà els dos valors corresponents a la mitjana i la variància.

```
def estadistica(llista):  
    m=mitjana(llista)  
    var=0  
    for x in llista:  
        var+=(x-m)**2  
    var/=(len(llista)-1)  
    return m,var
```

Exercici 6 (1 punt)

Fer un procediment anomenat `consultar_grup` que permeti fer consultes de les dades dels nadons que pertanyen a un determinat grup (A,B,C,D). El procediment rebrà com a paràmetres la llista amb els identificadors dels nadons, la llista amb les alçades dels nadons i la llista amb el grup al que pertanyen els nadons.

El procediment preguntarà quin grup (A, B, C o D) es vol consultar i a continuació mostrarà les dades (identificador i alçada) de tots els nadons que pertanyin al grup indicat. Si s'introdueix un grup inexistent, es mostrarà el missatge d'error: "Error: Grup inexistent" i es surt del procediment (NO es torna a preguntar el grup).

```
def consultar_grup(identificadors, alçades, grups):  
    consulta=input("Quin grup vols consultar? ")  
    if consulta not in ['A','B','C','D']:  
        print("Error: Grup inexistent")  
    else:  
        for i in range(len(grups)):  
            if grups[i]==consulta:  
                print(identificadors[i],alçades[i])
```

Exercici 7 (2 punts)

Fer un programa que segueixi els següents passos:

1. Inicialitzacions de variables i constants
2. Utilitzar (NO IMPLEMENTAR) el procediment `menu_principal()` que permet imprimir per pantalla el següent menú:

```
      ---- MENU ----  
1.- Introduir dades  
2.- Histograma d'alçades  
3.- Estadístiques  
4.- Consultar per grup  
5.- Finalitzar
```

3. Demanar a l'usuari que introdueixi una de les opcions.
4. Si l'opció és 1, cridar la funció de l'ex. 3, `gestionar_dades`, per a introduir les dades d'un nou nadó.
5. Si l'opció és 2:
 - 5.1. cridar la funció de l'ex. 4, `histograma`, per a generar l'histograma de les alçades dels nadons.
 - 5.2. mostrar quants nadons tenen cadascuna de les alçades amb missatges amb el següent format:

```
31 centímetres - X nadons  
32 centímetres - Y nadons  
...  
60 centímetres - Z nadons
```

on X serà el número de nadons amb 31cm, Y el número de nadons amb 32cm, etc.

6. Si l'opció és 3:
 - 6.1. comprovar si ja s'han introduït les dades d'algun nadó. En cas afirmatiu:
 - 6.1.1. cridar la funció de l'ex. 5, `estadística`, per a calcular les mesures estadístiques sobre les alçades dels nadons.
 - 6.1.2. mostrar un missatge amb la mitjana i la variància de les alçades dels nadons.
 - 6.2. si encara no s'han introduït dades, mostrar un missatge d'error.
7. Si l'opció és 4, cridar al procediment de l'ex. 6, `consultar_grup`, per a fer la consulta dels nadons que pertanyen a un determinat grup.
8. Si l'opció és 5, mostrar el missatge: `Sortint del programa...`
9. Qualsevol altra opció, escriure el missatge: `Opció no permesa.`
10. Repetir els passos 2 a 9, fins que l'opció del menú escollida sigui la 5.

```
AL_MIN=31
AL_MAX=60
identificadors=[]
alçades=[]
grups=[]

opcio='1'
while(opcio!='5'):
    menu()
    opcio=input("Introdueix una opció:")
    if (opcio=='1'):
        gestionar_dades(identificadors,alçades,grups)
    elif (opcio=='2'):
        hist=histograma(alçades)
        for i in range(AL_MIN,AL_MAX+1):
            print(i,"centímetres -",hist[i-AL_MIN],"nadons")
    elif (opcio=='3'):
        if (len(alçades)==0):
            print("Error: Encara no s'han introduït dades")
        else:
            mitj,var=estadistica(alçades)
            print("Mitjana -",mitjana,"Variància -",var)
    elif (opcio=='4'):
        consultar_grup(identificadors, alçades, grups)
    elif (opcio=='5'):
        print("Sortint del programa...")
    else:
        print("Opció inexistent")
```

Exercici 8 (1 punt)

Fer un programa que demani una cadena i a continuació mostri per pantalla cadascuna de les paraules que la formen en una línia diferent. Considereu que totes les paraules de la cadena estan separades entre elles per un caràcter espai ' '. No cal tenir en compte els signes de puntuació.

Per exemple, si la cadena que s'entra és "Hola, com esteu?" per pantalla s'hauria de mostrar:

Hola,
com
esteu?

```
cadena=input("Introdueix una cadena: ")
i=0
while i<len(cadena):
    1    paraula=""
        while i<len(cadena) and (cadena[i]!=" "):
            paraula+=cadena[i]
            i+=1
        i+=1
        print(paraula)
```

Exercici 9 (1 punt)

Quins seran els valors de x, y i z al finalitzar el programa?

```
def processar(x,w,llista):
    for i in range(w):
        llista.append(x)
        x+=w
    return x

def f(x,y,z):
    llista=[]
    z=processar(x,y,llista)
    x=len(llista)
    y+=1
    return(z,x,y)

x,y,z = f(2,3,2)
```

Valor x	Valor y	Valor z
11	3	4