

Fonaments de Programació

PAC3 - 20182

Data límit de lliurament: 18/03/2019

Estudiant

Cognoms: GALEANO SANCHO

Nom: DANIEL

Objectius

- Saber aplicar correctament l'estructura de control alternativa
- Aprendre a utilitzar el tipus de dades vector per representar estructures de dades senzilles.

Format i data de lliurament

Cal lliurar la PAC abans del dia **18 de març de 2019 a les 23:59**.

Per al lliurament caldrà que entregueu un fitxer en format ZIP, que contingui:

- Aquest document amb la resposta de l'exercici 1 i l'apartat b) de l'exercici 2
- El workspace de Codelite que contingui els fitxers .c demanats a l'exercici 2

Cal fer el lliurament a l'apartat de lliuraments d'AC de l'aula de teoria.

Enunciat

A la UOCAirways volen estudiar si la distribució de seients a un vol diari concret és rentable o no i ens han demanat la nostra ajuda per analitzar-ho.

El vol en qüestió té un total de 180 seients distribuïts actualment de la següent manera:

1. 16 seients de primera classe (4 seients per fila) amb un *pitch* (distància entre dos punts idèntics de dos seients en files consecutives) de 135 cm. i una llargada total per la primera classe de 6 metres. És a dir, hi ha un espai de 60 cm, entre la primera classe i la classe *business*.
2. 20 seients de classe business (4 seients per fila) amb un *pitch* de 100 cm. i una llargada total de 5,5 metres, és a dir que hi ha un espai de 50 cm de separació entre les classes *business* i *turista*.
3. 144 seients de classe turista (6 seients per fila) amb un *pitch* de 74 cm.

Ens proporcionen el següent algorisme que ens caldrà completar.

Exercici 1: Disseny en llenguatge algorísmic [50%]

Apartat a: [80%] Partint del següent esquelet d'algorisme, que ja té les constants declarades,

```
const
    NUM_SEATS_FIRST: integer = 16;
    NUM_SEATS_BUSINESS: integer = 20;
    NUM_SEATS_ECONOMY: integer = 144;
    PITCH_FIRST: integer = 135;
    PITCH_BUSINESS: integer = 100;
    PITCH_ECONOMY: integer = 74;
    LENGTH_FIRST: integer = 600;
    LENGTH_BUSINESS: integer = 550;
```

```
end const
```

```
algorithm simulator
```

```
var
    {Declaration of variables}
    ...
end var
...
end algorithm
```

completeu-lo fent els següents passos:

1. [10%] Declari un vector d'enters de tres posicions on guardar la mitjana de bitllets venuts durant l'últim any en cada classe. A la primera posició del vector hi va el nombre que correspon a la primera classe, a la segona posició del vector hi va la de classe *business* i a l'última posició del vector hi va la de la classe turista.
2. [10%] Declari altres variables i constants que puguin ser necessàries per resoldre el problema.
3. [10%] Llegeixi del canal estàndard d'entrada tres enters que corresponen a les tres mitjanes de bitllets venuts durant l'últim any (primera, *business* i turista). Per a cada lectura cal indicar quina informació es demana.
4. [10%] Comprovi que les dades entrades són correctes, és a dir que no són ni valors negatius ni superen el nombre màxim de seients de cada classe. Si les dades són incorrectes l'algorisme ha de donar un error i no continuar.
5. [60%] En cas contrari, és a dir, si les dades són correctes, faci el següent:
 - a. [40%] Si el % de la mitjana de bitllets venuts de primera classe és inferior al 25% dels seients disponibles en aquesta classe i el % de la mitjana de bitllets venuts en *business* és inferior al 50% dels seients disponibles en classe *business*, cal treure un missatge pel canal estàndard de sortida dient que es recomana convertir tot l'avió a una única classe, la turista, i calculi i mostri pel canal estàndard de sortida, quants seients nous de classe turista s'afegiran, si es treuen els seients de primera classe i els de *business*.
 - b. [40%] En cas contrari, si el % de la mitjana de bitllets venuts de primera és menor del 25% però el % de la classe *business* és superior o igual a 50%, cal mostrar un missatge pel canal estàndard de sortida que recomani treure la primera classe i reconvertir l'espai en classe *business*, i calculi i mostri pel mateix canal el número de seients de classe *business* que s'afegiran.
 - c. [20%] En cas contrari tregui un missatge pel canal estàndard de sortida indicant que cal fer un estudi amb més detall.

```
const
  NUM_SEATS_FIRST: integer = 16;
  NUM_SEATS_BUSINESS: integer = 20;
  NUM_SEATS_ECONOMY: integer = 144;
  PITCH_FIRST: integer = 135;
  PITCH_BUSINESS: integer = 100;
  PITCH_ECONOMY: integer = 74;
  LENGTH_FIRST: integer = 600;
  LENGTH_BUSINESS: integer = 550;
end const
```

```
algorithm simulator
```

```

var
    avgSold: vector[3] of integer;
end var

writeString("Enter last year sold tickets average for First Class: >> ");
avgSold [1] := readInteger();
writeString("Enter last year sold tickets average for Business Class: >> ");
avgSold [2] := readInteger();
writeString("Enter last year sold tickets average for Economy Class: >> ");
avgSold [3] := readInteger();

if (avgSold [1] > 0) and (avgSold [1] <= NUM_SEATS_FIRST) then
    if (avgSold [2] > 0) and (avgSold [2] <= NUM_SEATS_BUSINESS) then
        if (avgSold [3] > 0) and (avgSold [3] <= NUM_SEATS_ECONOMY) then
            if (avgSold[1] < 0.25*NUM_SEATS_FIRST) and (avgSold[2] <
0.5*NUM_SEATS_BUSINESS) then
                writeString("You should convert all the flight to economy class");
                writeString("Economy seats added: ");
                writeInteger( 6*(LENGTH_FIRST + LENGTH_BUSINESS)/PITCH_ECONOMY);
            else
                if (avgSold[1] < 0.25*NUM_SEATS_FIRST) and (avgSold[2] >=
0.5*NUM_SEATS_BUSINESS) then
                    writeString("You should convert all the first class seats to business class");
                    writeString("Business seats added: ");
                    writeInteger( 4*(LENGTH_FIRST)/PITCH_BUSINESS);
                else
                    writeString("You should do a more detailed study");
                end if
            end if
        else
            writeString("ERROR.");
        end if
    else
        writeString("ERROR.");
    end if
else
    writeString("ERROR.");
end if

end algorithm

```

Apartat b [20%] Explica què caldria modificar per fer aquest algorisme genèric. És a dir que no faci els càculs per un avió concret sinó que ho pugui fer per qualsevol vol. No cal fer el disseny, tant sols heu d'explicar que caldria fer.

Tan sols caldria declarar les constants com a variables, per a que es pugui demanar els seus valors concrets per a diferents vols. La resta de l'algorisme no faria falta modificar-lo.

Exercici 2: Programació en C [50%]

Apartat a: [70%] Implementeu en C l'algorisme de l'apartat a de l'exercici 1.

```
/*
** File: main.c
** Author: Daniel Galeano Sancho
** Date: 14-03-2019
** Description: PAC3
*/

/* System header file */
#include <stdio.h>

/* User defined types */
typedef enum {FALSE,TRUE} bool;

/* Main function */
int main(int argc, char **argv){

    /* Constants */
    const int NUM_SEATS_FIRST = 16;
    const int NUM_SEATS_BUSINESS = 20;
    const int NUM_SEATS_ECONOMY = 144;
    const int PITCH_FIRST = 135;
    const int PITCH_BUSINESS = 100;
    const int PITCH_ECONOMY = 74;
    const int LENGTH_FIRST = 600;
    const int LENGTH_BUSINESS = 550;

    /* Variables */
    int avgSold[3];

    printf("Enter last year sold tickets average for First Class: >> ");
    scanf("%d",&avgSold[0]);
    printf("Enter last year sold tickets average for Business Class: >>
");
    scanf("%d",&avgSold[1]);
    printf("Enter last year sold tickets average for Economy Class: >>
");
    scanf("%d",&avgSold[2]);

    if((avgSold[0]>0)&&(avgSold[0]<=NUM_SEATS_FIRST)){
        if((avgSold[1]>0)&&(avgSold[1]<=NUM_SEATS_BUSINESS)) {
            if((avgSold[2]>0)&&(avgSold[2]<=NUM_SEATS_ECONOMY)) {

                if((avgSold[0]<(0.25*NUM_SEATS_FIRST))&&(avgSold[1]<(0.5*NUM_SEATS_BU
SINESS))) {
                    printf("You should convert all the seats to
economy class.\n");
                    printf("Economy seats added:
%d.\n",6*(LENGTH_FIRST + LENGTH_BUSINESS)/PITCH_ECONOMY);
                }else{

                    if((avgSold[0]<(0.25*NUM_SEATS_FIRST))&&(avgSold[1]>=(0.5*NUM_SEATS_B
USINESS))) {

```

```

                printf("You should convert all the
first class seats to business class seats.\n");
                printf("Business seats added:
%d.\n", 4*(LENGTH_FIRST)/PITCH_BUSINESS);
            }else{
                printf("You should do a more detailed
study.\n");
            }
        }
    }else{
        printf("ERROR.\n");
    }
}else{
    printf("ERROR.\n");
}
return 0;
}

```

Apartat b: [30%] Com a les anteriors PAC, es demana que indiqueu jocs de prova per provar l'algorisme. Concretament, ompliu la taula següent:

Input data			Output data
average1	average2	average3	
17	15	140	ERROR
3	18	130	You should convert all the first class seats to business class seats. Business seats added: 24
3	8	130	You should convert all the seats to economy class. Economy seats added: 93
5	12	120	You should do a more detailed study.

On *average1*, *average2* i *average3* són els valors que cal entrar al programa.

Criteris de correcció:

A l'exercici 1:

- Que se segueixi la notació algorísmica utilitzada a l'assignatura. Vegeu document Nomenclator a la xWiki de contingut.
- Que se segueixen les instruccions donades i l'algorisme respongui al problema plantejat.
- Que s'utilitzi correctament l'estructura alternativa, el tipus de dades vector i, si s'escau, les funcions de conversió de tipus.
- Que es raoni correctament la resposta de l'apartat b de la primera pregunta.

A l'exercici 2:

- Que el programa s'adeqüi a les indicacions donades.
- Que el programa compila i funciona d'acord amb el que es demana.
- Que es respectin els criteris d'estil de programació C. Vegeu la Guia d'estil de programació en C que teniu a la Wiki de contingut.