

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

AgroFruts, una cooperativa agrícola, vol emmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardara en tres taules (arrays) (una per els identificadors, una per les mesures dels diàmetres i una per als nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix índex de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà és inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$35 + 40 = \underline{75}$$

$$\begin{array}{r} 25 - 3 = 22 \\ 1 + 1 = 2 \\ \hline \text{RESULTAT : } 24 \end{array}$$

Heu una funció anomenada `QualitatLot` que veurà els valors de diàmetre mitjà i nivell de qualitat i els guardara en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornara -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introduixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardarà a la posició indicada de les taules corresponents.

En els dos casos, si s'introduix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el nivell de qualitat del lot (utilitzant la funció QualitatLot de l'ex.1) i es guardará a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

Fer un procediment anomenat DistribucioPerNivells que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)

Fer una funció anomenada Estadistica que calculi i retorna alguns estadístics (mitjana i variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retornarà 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:
$$\text{variància} = \frac{1}{N-1} \sum_i (x_i - \mu)^2$$
 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.

Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

Exercici 4 (1 punt)

Fer un procediment anomenat ConsultaPerNivell que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguin el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.6 cms").

Si s'introdueix un nivell de qualitat que no és vàlid (nivells vàlids: 1, 2, 3, 4), es mostrerà el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que tens implementat un procediment definit com:

```
void InicialitzarTaula (int taula, int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribuacioPerNivells de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (3 punts)

Declarar un nou tipus de dades, TIOT, com un registre amb els camps: `Ident_Lot` (valor enter), `Diametre` (valor real), `Qualitat` (valor enter) i `Origen_Lot` (cadena de 50 caràcters). Fer un procediment anomenat `OmplirTIOT` que rebi com a paràmetre un registre del tipus TIOT i ompli els seus camps a partir de la informació entrada per teclat.

Exercici 9 (1 punt)

Fer una funció anomenada `BuscarPrimerZero` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat `QuadratMatriu` que elevi al quadrat tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

Note: Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`

VOTS →

0	1	2	3
1	2	1	0

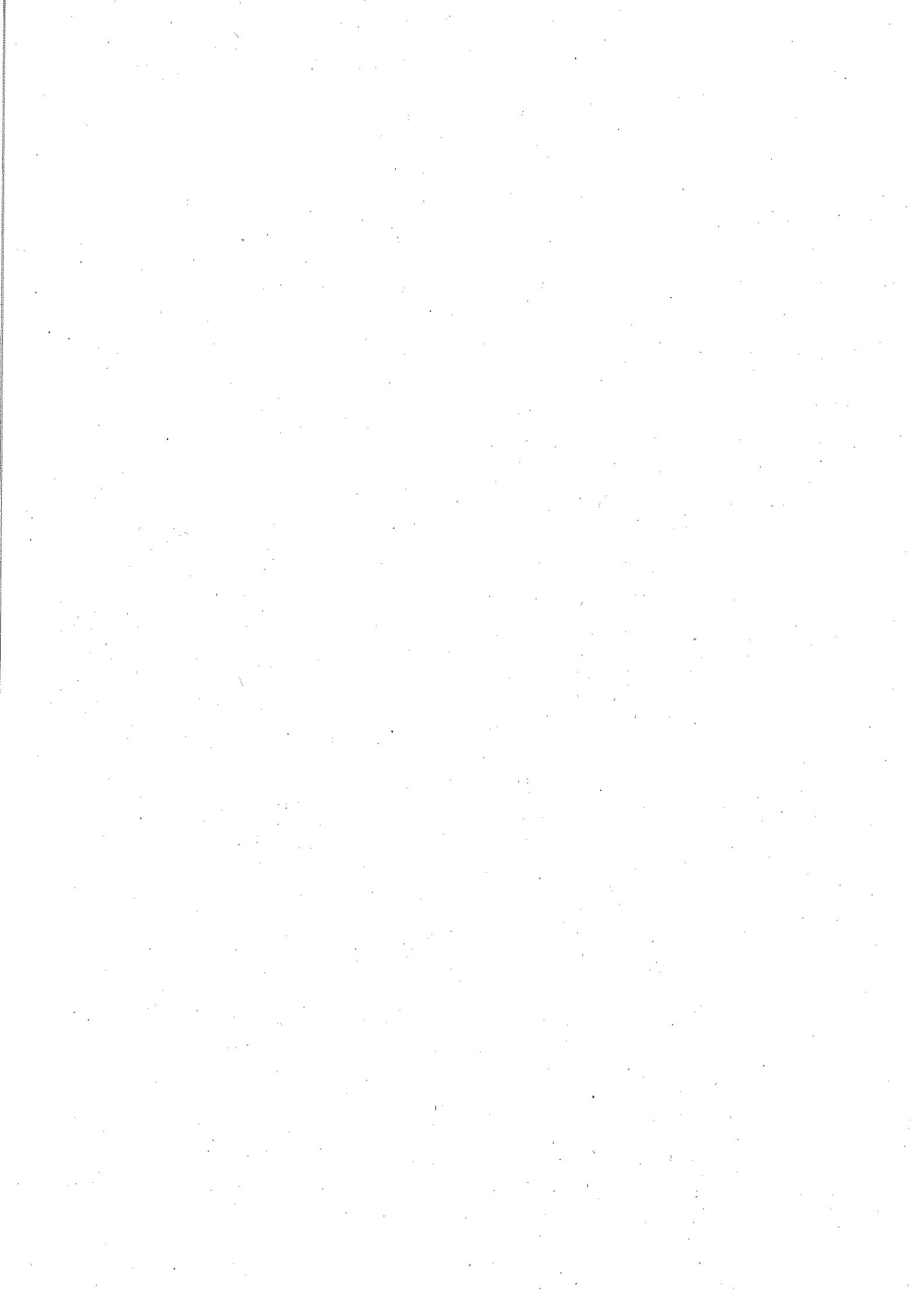
134

RESULTS →

0	1	2
1	2	0

134

6304



Examen Parcial (11 de Novembre de 2019)

Grup: 1230293
NIU:

Nom estudiant: VAN SCHAESSEN

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

AgroFruits, una cooperativa agrícola, vol emmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (entre 1 i per a cada lot) i per a cada lot és vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardara en tres taules (arrays) una pels identificadors, una pels diàmetres i una pels nivells de qualitat. La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix index de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà és inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$7 \times 7 = 49$$

$$10 \times 10 = \boxed{100}$$

Per una funció anomenada `mitjana` que calcula la mitjana de tres valors introduïts per teclat, se li passaran tres valors i es retornarà la mitjana. Les tres taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introdueixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introdueix un valor que no es positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anterior, es calcularà el nivell de qualitat del lot (utilitzant la funció QualitatLot de l'ex.1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$68 + 2 = 70$$

$$50 + 50 = 100$$

Resultat: 170

Exercici 4 (1 punt)
Fer un procediment anomenat `DistribucioPerNivells` que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) està inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)
Fer una funció anomenada `Estadistica` que calculi i retorni alguns estadístics (mitjana i variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variància, la funció retornara 0.

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retornara 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:

$$\text{variància} = \frac{1}{N-1} \sum_i (x_i - \mu)^2 \quad \text{on } x_i \text{ són els elements del conjunt de dades, } \mu \text{ és la mitjana de les dades i } N \text{ és el nombre d'elements del conjunt de dades.}$$

Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`.

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerNivell` que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguin el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.6 cms").

Si s'introdueix un nivell de qualitat que no és vàlid (nivells vàlids: 1, 2, 3, 4), es mostrerà el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula, int nElements);
```

que permet inicialitzar a zero una taula d'enteros de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment entrardadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introduixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerNiveus de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
- 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil:
"Nivell 1: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Declarar un nou tipus de dades, TIOT, com un registre amb els camps: Ident_lot (valor enter), Diametre (valor real), Qualitat (valor enter) i Origen_lot (cadena de 50 caràcters).

Fer un procediment anomenat OmplirIOT que rebi com a paràmetre un registre del tipus TIOT i ompli els seus camps a partir de la informació entrada per teclat.

Exercici 9 (1 punt)

Fer una funció anomenada BuscarPrimerZero que rebi com a paràmetres un vector d'enter i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retornar -1.

Exercici 8 (1 punt)

Fer un procediment anomenat QuadratMatriu que elevi al quadrat tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

Note: Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

int ComprobacionPrestes (int gd, int sa, int vg)

{

 cin >> gd;

 cin >> sa;

 cin >> vg;

 tar puntST;

 int puntos1 = 0;

 int puntos2 = 0;

 if (sa >= (gd / 2))

 puntos1 = puntos1 + 5;

 else

 if ((sa < (gd / 2)) && (sa >= (gd / 4)))

 puntos1 = puntos1 + 3;

 else

 if ((sa < (gd / 4)) && & (sa > (gd / 10)))

 puntos1 += 1;

 if (vg >= (2 * sa))

 puntos2 = puntos2 + 5;

 else

 if ((vg > (gd / 2)) && & (vg <= (gd * 2))).

 puntos2 = puntos2 + 3;

puntST = puntos1 + puntos2;

cout << puntST;

typef struct

Examen Parcial (11 de Novembre de 2019)

Grup: 3875067
NIU: 3875067

Nom estudiant: BOB BY ALEXANDRA

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

AgroFruits, una cooperativa agrícola, vol emmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardarà en tres taules (arrays) (una per als identificadors, una per les mesures dels diàmetres i una per als nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix index de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà és inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà es superior o igual a 9 cms.

NOTA: Considerau que els valors d'entrada són positius (no cal fer la comprovació).

$$2 + 2 = 4$$
$$3 + 3 = \underline{6}$$

$$90 \times 90 = 8.100$$

$$7 + 7 = 14$$

RESULTAT : 8.114

guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index valid de les taules). Si la posició és incorrecta, la funció retornara -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introdueixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introdueix un valor que no és positiu, es mostrarà el missatge "Error: Valor Incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el nivell de qualitat del lot (utilitzeu la funció QualitatLot de l'ex.1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

Fer un procediment anomenat `DistribucioPerNivells` que calculi quants lots hi ha de cada nivell de qualitat ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) està inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 3 (1 punt)
Fer una funció anomenada `Estatistica` que calculi i torni alguns estadístics (mitjana, variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variància dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retornarà 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:

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

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.

Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreta matinh.

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerNivell` que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrara per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguin el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.5 cms"). Si s'introdueix un nivell de qualitat que no és vàlid (nivells vàlids: 1, 2, 3, 4), es mostrerà el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

Fer un programa complet (declaracions globals i funció main) que segueixi els següents passos.

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on corresponguï) una estructura de ciències anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula, int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula de distribució.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerNivells de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 lots."
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Declarar un nou tipus de dades, TIOT, com un registre amb els camps: Ident_Lot (valor enter), Diàmetre (valor real), Quantitat (valor enter) | origen_Lot (cadena de 50 caràcters). Fer un procediment anomenat AmplILOT que rebi com a paràmetre un registre del tipus TIOT i ompli els seus camps a partir de la informació entrada per teclat.

Ejercicio 3 (1 punto)
Fer una funció anomenada BuscarPrimerZero que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retornarà -1.

6.

```

int IndexMinNoZeroArray (int v[], int dim) {
    int i, minim, indexmin;
    while (v[i] == 0) {
        i++;
    }
    minim = v[i];
    indexmin = i;
    i++;
    while (i < dim || minim != 1) {
        if (v[i] < minim && v[i] != 0) {
            minim = v[i];
            indexmin = i;
        }
        i++;
    }
    return indexmin;
}

```

do {

```

    jet = Results(i)
    i++
    while (i < dipotats + 1 || jet == 0)

```

$\cos \left[\sqrt{V_0} \right] +$

$1 / 2 / 5 / 6 / 9 / 5 / 2 / 1$

$1 / 6 / 5 / 3 / 6 / 8 / 1 / 0 / 1$

Examen Parcial (11 de Novembre de 2019)

Grup: S769112

Nom estudiant:

CHRISTOPHER RODRIGUEZ NIU:

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

AgroFruits, una cooperativa agrícola, vol emmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardarà en tres taules (arrays) (una pels identificadors, una per les mesures dels diàmetres i una pels nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix index de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms), i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà es inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$80 \times 10 = \underline{800}$$
$$12 + 12 = \underline{24}$$

$$60 + 10 = 70$$

$$80 + 10 = 90$$

RESULTAT : 100

guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introdueixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introduïx un valor que no es positiu, es mostrerà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anterior, es calcularà el nivell de qualitat del lot (utilitzant la funció QualitatLot de l'ex.1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

Fer un procediment anomenat `DistribuacioPerNiveis` que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer parametre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guarda el número de lots de cada nivell (tercer paràmetre) està inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 4 (1 punt)
Fer un procediment anomenat `ConsultaPerNivell` que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguen el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.6 cms").

Si s'introdueix un nivell de qualitat que no és vàlid (nivells vàlids: 1, 2, 3, 4), es mostrara el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

Exercici 5 (1 punt)
Fer una funció anomenada `Estatistica` que calculi i retorna alguns estadístics (mitjana, variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variància dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retorna à 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:

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

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.

Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`.

Fer un programa complet (declaracions globals i funcions utilitzades) que implementa una distribució de tots els llets.

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula, int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilitzau (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerNivells de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 lots."
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Declarar un nou tipus de dades, `Tilot`, com un registre amb els camps: `Ident_lot` (valor enter), `Diametre` (valor real), `Qualitat` (valor enter) i `Oxigen_lot` (cadena de 50 caràcters).
Fer un procediment anomenat `ComplirLot` que rebi com a paràmetre un registre del tipus `Tilot` i ompli els seus camps a partir de la informació entrada per teclat.

Fer una funció anomenada `BuscarPrimerZero` que rebi com a paràmetres un vector, d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat `QuadratMatrui` que elevi al quadrat tots els elements d'una matrui d'enters de n files i 5 columnes. El procediment rebra com a paràmetres: una matrui d'enters de 5 columnes, el número de files de la matrui, i el número de columnes de la matrui.

Note: Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`

anaments d'Informàtica (103806)

Examen Parcial (11 de Novembre de 2019)

Nom estudiant: PAMELA ZUBIA ALV

Curs 2019-2020
Grup:
NIU: 3668464

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) una per els mesures del número de pètals i una pel codi de venda. La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix index de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$40 + 50 = 90$$

$$73 + 13 = \boxed{26}$$

$$100 + 100 = 200$$

$$17 + 10 = 27$$

RESULTAT : 227

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introduceix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introdueixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzant la funció CodiVendaLot de l'ex. 1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucióPerCodi` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada codi de venda en una posició different.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i número mitjà de pètals) dels lots que tinguen el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.6").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrara el missatge "Error: Codí de venda inexistent" i es sortirà.

Exercici 5 (1 punt)

Fer una funció anomenada `Estadística` que calculi i retornei alguns estadístics (màxim, mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

atenció (12.0 punts)

- Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:
1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
 2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
 3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.
 4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula(int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.

 5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
 6. Mostrar un missatge indicant que té 4 opcions disponibles ({-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir}) i demanar a l'usuari que introdueixi una de les opcions.
 7. Llegir l'opció escollida.
 8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codí 10: 12 lots" 9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodi de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
 10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
 11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
 12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.5 punts)

Declarar un nou tipus de dades, TLot, com un registre amb els camps: Ident_lot (valor enter), Num_mitja_petals (valor real), Codi_venda (valor enter) i Origen_lot (cadena de 50 caràcters).

Fer un procediment anomenat InicialitzarLot que rebi com a paràmetre un registre del tipus TLot i inicialitzi els seus camps amb valor zero als camps numèrics, i valor "...." al camp de tipus cadena de caràcters.

Exercici 9 (1 punt)

Fer una funció anomenada BuscarPrimerImparell que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornara -1.

Exercici 8 (1 punt)

Fer un procediment anomenat DobleMatriu que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Nom estudiant:

Etkel Mccallay Crespo

Grup:

NIU:

210 66 37

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot és vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una per els identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$42 - 20 = 22$$
$$22 + 3 = \underline{\underline{25}}$$

Exercici 2 (1 punt)

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en tres taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introdueix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornara a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzeu la funció CodiVendaLot de l'ex. 1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$31 \times 3 = 93$$

$$93 + 100 = 193$$

Resultat : 193

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodi` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada codi de venda en una posició diferent.

NOTA: La taula on es guardaranà el número de lots de cada codi de venda (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retornei alguns estadístics (màxim i mínim), calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el codi de venda que vol consultar. A continuació, el procediment mostrerà per pantalla les dades (identificador i número mitjà de pètals) dels lots que tinguen el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.5").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codi de venda inexistent" i es sortirà.

Exercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enteros de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribuacioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codí 10: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodi de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.5 punts)

Declarar un nou tipus de dades, `TIot`, com un registre amb els camps: `Ident_Lot` (valor enter), `Num_mitja_petals` (valor real), `Codi_venda` (valor enter) i `Origen_lot` (cadena de 50 caràcters).
Fer un procediment anomenat `IniciarLot` que rebi com a paràmetre un registre del tipus `TIot` i inicialitzi els seus camps amb valor zero als camps numèrics, i valor “_” al camp de tipus cadena de caràcters.

Exercici 9 (1 punt)

Fer una funció anomenada `BuscarPrimerImparell` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornar -1.

Exercici 8 (1 punt)

Fer un procediment anomenat `DoblaMatriu` que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

1530245

int i = 0;
 odd zero == false
 while (~~(v[i] != 0 & i < dim) &~~
~~zero == false~~ {
 zero == true;
 if (v[i] == 0) zero = true;
 }

if he ist
~~v[i]~~ = v[i];
 1 - 35
 position
 zu weit.

int INNZA (int v[], int dim);

int minimum = 0; if $\text{minimum} < \text{dim}$:
 while ($v[\text{minimum}] == 0$):

minimum++;

}

for (int i = minimum; i < dim; i++) {

if ($v[i] > v[\text{minimum}]$)

minimum = i;

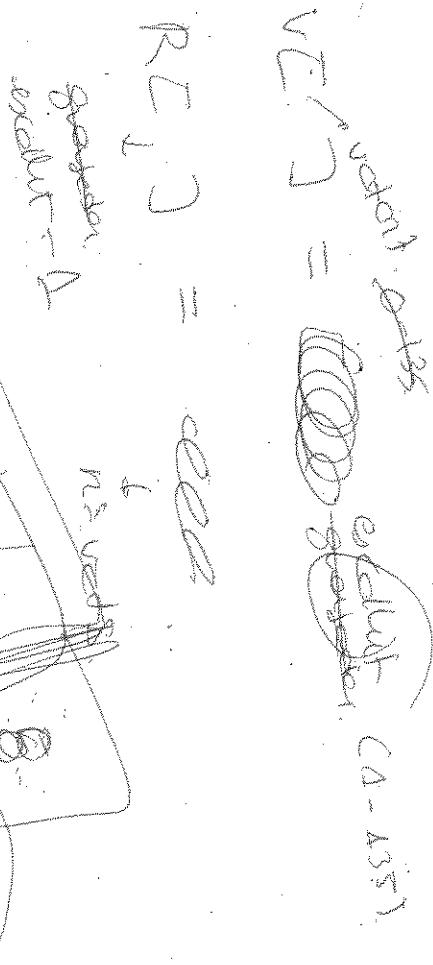
}

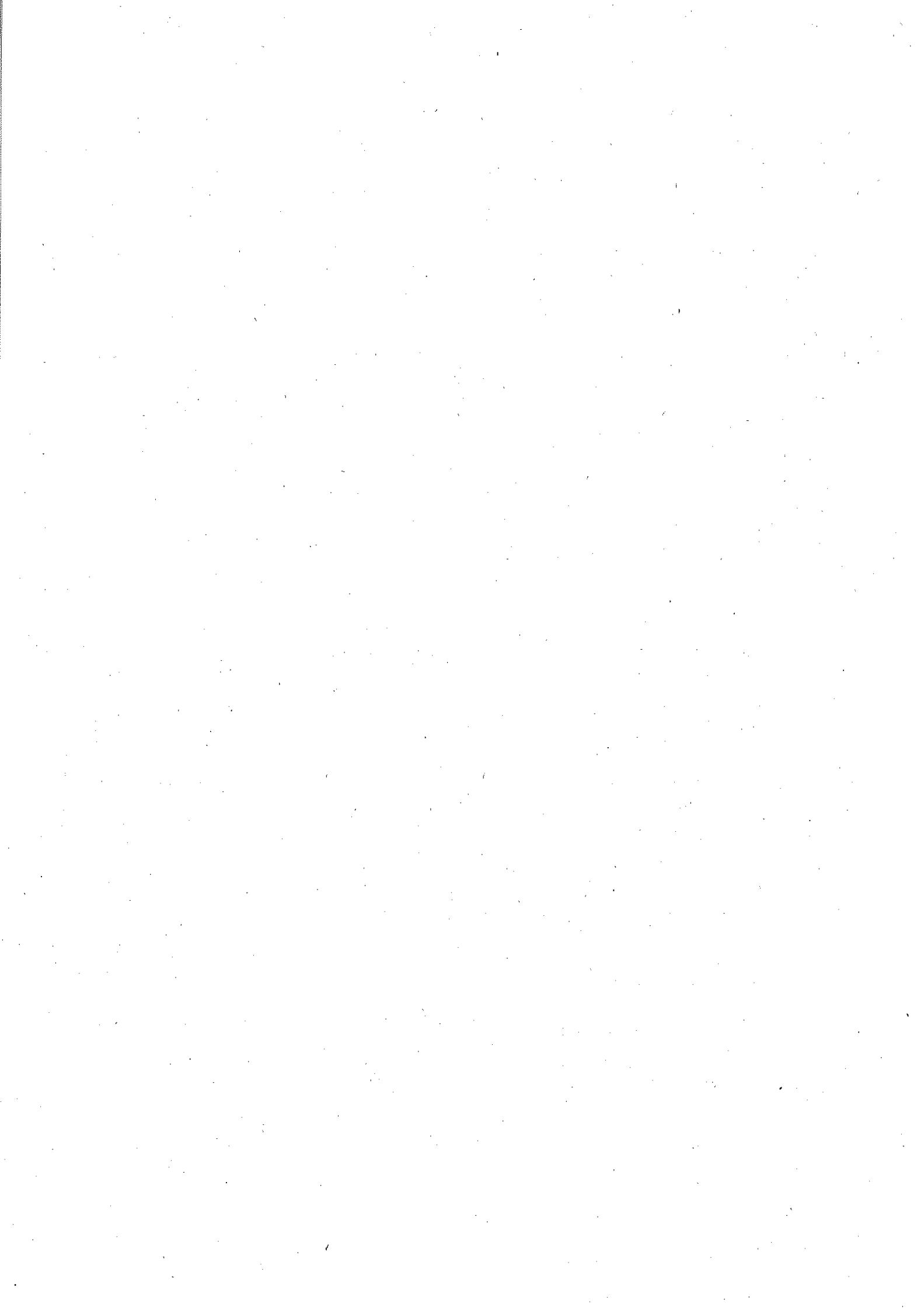
return minimum;

}

$v[0] = 2$
 $/v[i-1]$

101010101
 0 1 2 3 4





Fòrnaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: NIU: 8331131

Nom estudiant: Cody Jones Henderson

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a constitució:

AgroFruits, una cooperativa agrícola, vol emmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardara en tres taules (arrays) una per als identificadors, una per les mesures dels diàmetres i una per als nivells de qualitat. La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix index de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà és inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$20 \div 10 = 2 \\ 2 \times 4 = \boxed{8}$$

$$81 + 9 = 89$$

$$89 - 19 = 70$$

Resultat: 70

Exercici 2 (1 punt)
Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de pomes i les guardará en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introdueixin

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició es incorrecta, la funció retornara -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introduixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introdueix un valor que no és positiu, es mostrerà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el nivell de qualitat del lot (utilitzant la funció QualitatLot de l'ex.1) i es guardará a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerNivells` que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerNivel` que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanara a l'usuari que introduixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrà per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguin el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostren en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.6 cms").

Si s'introdueix un nivell de qualitat que no és vàlid (nivells vàlids: 1, 2, 3, 4), es mostrerà el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorni alguns estadístics (mitjana i variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variància dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retornarà 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:

$$\text{variància} = \frac{1}{N-1} \sum_i (x_i - \mu)^2 \quad \text{on } x_i \text{ són els elements del conjunt de dades, } \mu \text{ és la mitjana de les dades i } N \text{ és el nombre d'elements del conjunt de dades.}$$

Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor, 2)` de la llibreria `math.h`.

Exercici 6 (2.5 punts)
Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula, int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerNivells de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 Lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadistics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.5 punts)

Declarar un nou tipus de dades, `tLot`, com un registre amb els camps: `Ident_lot` (valor enter), `Diametre` (valor real), `Qualitat` (valor enter) i `Origen_lot` (cadena de 50 caràcters). Fer un procediment anomenat `ComplirLot` que rebi com a paràmetre un registre del tipus `tLot` i omplí els seus camps a partir de la informació entrada per teclat.

Exercici 9 (1 punt)

Fer una funció anomenada `BuscarPrimerZero` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retorna -1.

Exercici 8 (1 punt)

Fer un procediment anomenat `QuadratMatrxiu` que elevi al quadrat tots els elements d'una matrxiu d'enters de `n` files i 5 columnes. El procediment rebrà com a paràmetres: una matrxiu d'enters de 5 columnes, el numero de files de la matrxiu, i el numero de columnes de la matrxiu.

Note: Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`

Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: 8836355

Nom estudiant: Jason Hawn Ryan Haugen

Exercici 2 (1 punt)
Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$125 + 4 = \boxed{129}$$

$$25 \times 2 = 50$$

$$60 - 10 = 50$$

Resultat: 50

En els dos casos, si s'introdueix un valor que no és positiu, es mostrarà el missatge "Error: Valor Incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzant la funció CodiVendaLot de l'ex.1) i es guardara a la taula correspondent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodi` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada codi de venda en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorni alguns estadístics (màxim i mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanara a l'usuari que introdueixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i número mitjà de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.5").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codi de venda inexistent" i es sortirà.

Exercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribucio per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codí 10: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodí de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.5 punts)

Declarar un nou tipus de dades, TIot, com un registre amb els camps: Ident_Lot (valor enter), Num_mitja_petals (valor real), Codi_venda (valor enter) i Origen_Lot (cadena de 50 caràcters).

Fer un procediment anomenat InicialitzarLot que rebi com a paràmetre un registre del tipus TIot i inicialitzi els seus camps amb valor zero als camps numèrics, i valor “---” al camp de tipus cadena de caràcters.

Exercici 9 (1 punt)

Fer una funció anomenada BuscarPrimerImparell que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat DobleMatríu que doble el valor de tots els elements d'una matríu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matríu d'enters de 5 columnes, el número de files de la matríu, i el número de columnes de la matríu.

Menu principal () :

cn>> opua;

Switch (opua)

case 1 : while((Identificador < 0) || (Identificador > 135))

cout << "Digieren el seu Identificador " << endl;

(cn>> Identificador);

cout << "Diguem els vots del minum i maxim " << endl;

(cn>> min >> max);

vot = Voter (min, max);

Vots [N diputats] = vot;

+ ° 0°'5

10 → 10°

→ 70

700/100

7

10 · 0'7 = 7

I mit punts = 0;

I g (salari == preter * 0'50)

Salari = demande * 0'50

at, 0

52,0 - 0)

300

2000

$$\frac{52}{10} = \frac{52}{100}$$

$$= \frac{52}{100}$$

- 000.

$$\text{parallel} = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2}$$

$$Voh [CI] \% = \frac{\text{Realkohol} [CI]}{\text{Realkohol} [CI] + \text{Methylalkohol} [CI]}$$

$$= \{ \text{Realkohol} [CI] \}$$

$$g_{\text{so}} (\text{Realkohol} [CI]) = 100 \cdot 10^{-3}$$

Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup:

Nom estudiant: William Odilex Roseville NIU: 5481633

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter) i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una per els identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix index de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals es superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$150 + 60 = \underline{210}$$

Resultado : 12

$$35 + 8 = 43$$

$$4 \times 3 = 12$$

Exercici 2 (1 punt)
Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introdueixin
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introdueixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introduceix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzeu la funció CodiVendaLot de l'ex. 1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodi` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre); i una taula on es guardarà el número de lots de cada codi de venda en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estarà inicializada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorni alguns estadístics (màxim i mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

Exercici 4 (1 punt)
Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introdueixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i número mitjà de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.6").

Si s'introdueix un codi de venda que no es valid (codis valids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codi de venda inexistent" i es sortirà.

Exercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribucio per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula(int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodis de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0,5 punts)

Declarar un nou tipus de dades, `TLot`, com un registre amb els camps: `Ident_lot` (valor enter), `Num_mitja_petalys` (valor real), `Codi_venda` (valor enter) i `Origen_lot` (cadena de 50 caràcters).

Fer un procediment anomenat `InicialitzarLot` que rebi com a paràmetre un registre del tipus `TLot` i inicialitzi els seus camps amb valor zero als camps numèrics, i valor `"___"` al camp de tipus cadena de caràcters.

Exercici 9 (1 punt)

Fer una funció anomenada `BuscarPrimerImparell` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat `DobleMatriu` que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup:

NIU: 2535847

Nom estudiant: Gwen Brooks King

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenu a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una pels identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$125 + 30 = 155$$
$$200 - 70 = \underline{130}$$

Exercici 2 (1 punt)

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornara -1 i no es farà res més.

Si la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardaran a la posició indicada de les taules corresponents.

En els dos casos, si s'introduceix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzau la funció CodiVendaLot de l'ex.1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$15 \times 2 = 30$$

$$20 + 20 = 40$$

Resultat: 40

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodi`s que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada codi de venda en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estàrà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorna alguns estadístics (màxim i mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i número mitjà de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.e.: "ID: 1234 num mitja de petals: 32.6").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrerà el missatge "Error: Codí de venda inexistent", es sortirà.

Exercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula(int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístiques, 4-Sortir) ; demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodi de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.5 punts)

Declarar un nou tipus de dades, `TLot`, com un registre amb els camps: `Ident_lot` (valor enter), `Nom_mitja_petals` (valor real), `Codi_venda` (valor enter) i `Origen_lot` (cadena de 50 caràcters).

Fer un procediment anomenat `InicialitzarLot` que rebi com a paràmetre un registre del tipus `TLot` i inicializi els seus camps amb valor zero als camps numèrics, i valor "" al camp del tipus cadena de caràcters.

Exercici 9 (1 punt)

Fer una funció anomenada `BuscarPrimerImparell` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat `DoblaMatriu` que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

3. int Voter (int min; int max)

int i;

for (i = min; i <= max; i++)

~~vector~~

int m;

while ($N < \text{min}$ || $N > \text{max}$)

cout << "Error"

cin >> m;

void InitializeArray (VC[], DIM, Valor)

400

100

NozeroArray (int VC[], int DIM)

bool trobat = false; if (i == 0)

int m;

valors (cin >> m)

while ($i < \text{DIM}$) || !trobat)

class VC

float NormalVector (int x1, int x2, int y1, int y2);
float resultat;

int x1, int x2, int y1, int y2;

resultat = float sqrt (pow(x2 - x1) + pow(y2 - y1))

return resultat;

do

cin >> ~~VC~~ []

if (VC[] == 0)

trobat = true

} while ($i < \text{DIM}$ || !trobat)

if (trobat)

return i;

else

return 0;

vDemocrat > 2. Salari

(Salari, 2. vDemocrat)

return 5;

(vDemocrat <

(Salari > 4 vDemocrats)

ff (vDemocrat > 2.5 Salari)

2

Salari = $\frac{1}{2}$ prestec

2 Salari = prestec

Salari \geq prestec / 2

Salari \geq prestec / 4

ff Salari \leq prestec / 2

void Escalate (Votes & M, Diw, ~~selected~~ results C3)

Votes C3

!<= !&

for (C<0; i<Diw; i++)

Results C3 = Votes C3

if (!!(int) V[i] < V[i-1])

int maxRow, pos min;

int IndexMinZeroZero (array (VC3, Diw))

int min = VC3[0];

if (V[i] == 0) !<Diw; i++)

for (i=0; i<Diw; i++)

int maxRow, pos min;

int IndexMinZeroZero (array (VC3, Diw))

Results C3 = Votes C3

for (C<0; i<Diw; i++)

!<= !&

Votes C3

void Escalate (Votes C3, Diw) {

case 3: ;

case 2: ;

case 1: ;

switch (min)

case 1: ;

new Pmapupar (

InitiateAutotransfer (Votes C3, Diw, m)

int min;

int m;

int Diw;

int Resultats C3;

int Votec3;

int RuleIndex, resus

return maxRow;

3

int maxRow = VC3[0];

if (V[i] < V[i-1])

for (i=0; i<Diw; i++)

int maxRow, pos min;

int IndexMinZeroZero (array (VC3, Diw))

Results C3 = Votes C3

for (C<0; i<Diw; i++)

!<= !&

Votes C3

Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)
Nom estudiant: Rita Sílvia Marcos

Grup: 2 U 19 780
NIU:

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenui a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una per als identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix index de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Per una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$\begin{array}{r} 200 - 50 = 150 \\ 320 + 4 = \boxed{324} \end{array}$$

Resultado: 60

$$12 + 15 = 27$$

$$30 \times 2 = 60$$

Exercici 2 (1 punt)
Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.
Si la posició és correcta, es demanarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardaràn a la posició indicada de les taules corresponents.
En els dos casos, si s'introduceix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introdueixin els valors correctament.
Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzant la funció CodiVendaLot de l'ex.1) i es guardarà a la taula corresponent.
Si s'han introduït totes les dades correctament, la funció retornarà 0.

Exercici 3 (1 punt)

Fer un procediment anomenat DistribucioPerCodi s que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardara el numero de lots de cada codi de venda en una posició diferent.

NOTA: La taula on es guardara el número de lots de cada codi de venda (tercer paràmetre) estarà inicializada a 0 (no cal inicialitzar-la en aquest procediment).

Initialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)

Fer una funció anomenada Estadistica que calculi i retorna alguns estadistics (màxim i mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadistics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

Exercici 4 (1 punt)

Fer un procediment anomenat ConsultaPerCodi que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el codi de venda que vol consultar. A continuació el procediment mostrarà per pantalla les dades (identificador i número mitjà de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.6").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codi de venda inexistent" i es sortirà.

Exercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilieu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.
5. Utilizar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodis de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0,5 punts)

Declarar un nou tipus de dades, TIot, com un registre amb els camps: Ident_Lot (valor enter), Num_mitja_petals (valor real), Codi_venda (valor enter) i Origen_lot (cadena de 50 caràcters).

Fer un procediment anomenat InicialitzarIot que rebi com a paràmetre un registre del tipus TIot i inicialitzi els seus camps amb valor zero als camps numèrics, i valor "" al camp de tipus cadena de caràcters.

Exercici 9 (1 punt)

Fer una funció anomenada BuscarPrimerImparell que rebi com a paràmetres un vector d'enters i un valor enter amb el número dels elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat DobleMatriu que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

45375601

```

for (i=0; i<dim; j++)
{
    int minm;
    absit
    if (a[i]<minm) && (a[i] != 0)
        al[i]=minm;
}
return i;

```

~~if (a[i]<a[i+1]) && (a[i] != 0)~~
~~a[i]=minm~~

for (i=0; i<N-PAR1; i++)
 if (vote[i]<=

```

int word (int min, int max)
{
    cout << "Enter ";
    cin >> w;
    cout << "Enter ";
    cin >> l;
    cout << "Enter ";
    cin >> r;

    if (w <= l && l <= r)
        cout << "Valid Range" << endl;
    else
        cout << "Invalid Range" << endl;
}

int sum (int n)
{
    int sum = 0;
    for (int i = 1; i <= n; i++)
        sum += i;
    return sum;
}

int main ()
{
    cout << "Enter ";
    cin >> n;
    cout << "Sum of first " << n << " numbers is " << sum (n) << endl;
}

```

Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: 1457091

Nom estudiant: Janet Young Hebeat

NIU: 1457091

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenui a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter) i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una per als identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el numero mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el numero mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$200 + 100 = \underline{300}$$
$$100 - 50 = \underline{150}$$

Exercici 2 (1 punt)
Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introduceix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introdueixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzant la funció CodiVendaLot de l'ex.1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$15 - 3 = 12$$

$$20 \times 7 = 140$$

Resultado : 140

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodi` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer parametre), i una taula on es guardarà el número de lots de cada codi de venda en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i torni alguns estadístics (màxim i mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i número mitjà de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.6").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: codi de venda inexistent" i es sortirà.

Exercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on corresponguï) una estructura de dades anomenada distribucio per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodi de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0,5 punts)

Declarar un nou tipus de dades, TIot, com un registre amb els camps: Ident_Lot (valor enter), Num_mitja_Petals (valor real), Codi_venda (valor enter) i Origen_Lot (cadena de 50 caràcters).
Fer un procediment anomenat InicialitzarLot que rebi com a paràmetre un registre del tipus TIot i inicialitzi els seus camps amb valor zero als camps numèrics, i valor "...." al camp de tipus cadena de caràcters.

Exercici 9 (1 punt)

Fer una funció anomenada BuscarPrimerImparell que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat DobleMatriu que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

152926
Jueves 26/07/2018
Ticatá
define max 155
define min -1

NoZeroArray
Escritura -
NuevoPrincipal

Individuo
IndividuoArray
Individuo

IndividuoArray
Individuo

Individuo
Individuo

```
int main () {  
    int opcion; int nPart; int Vots[max]; int Resultado[min]; int i;  
    int nPart; int Vots[max]; int Resultado[min]; int i;  
    int nPart; int Vots[max]; int Resultado[min]; int i;
```

```
    do {  
        NuevoPrincipial();
```

```
        cin >> opcion;
```

```
    } while (opcion != 0);
```

```
    case 1: do { cin >> nPart; } while ((nPart<1) || (nPart>35));
```

```
    Votar(min, max);
```

```
    break;
```

```
    case 2: NoZeroArray(Vots, max);
```

```
    Escritura(Vots, max, Resultado);
```

```
    break;
```

```
    case 3: IndividuoArray(Vots, max, Resultado);  
    for (i=0; i<max; i++) { cout << Resultado[i] << " " ; }
```

```
int Voter(int min, int max)
{ int vot;
```

args.vot

if(vot > max) { vot = max }

do { vot = args.vot + 1; if(vot <= max) { break; } }

args.vot =

if(vot < min) { vot = min } cout << "Error: position must be between " << min << " and " << max << endl;

return vot;

```
while((x < Dem) & (y < Prob)) { c = A * (prob - x) + b * (dem - y); return
```

```
int comprensioPreStoc( float demand, float salary, float competads) {
    int punts;
```

```
if(salary >= demand/2) { punts = punts + 5; } else if(salary > demand/4) { punts = punts + 3; } else if(salary < demand/4) { punts = punts + 1; } else { punts = punts + 0; }
```

```
if(proficiat > 2 * demand) { punts = punts + 5; } else if(proficiat < 2 * demand) & & (proficiat < demand) { punts = punts + 3; } else { punts = punts + 0; }
```

return punts;

Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: NIU: 1386705

Nom estudiant: 237425025

Important. Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherentia entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

AgroFruits, una cooperativa agrícola, vol emmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardara en tres taules (arrays) (una pels identificadors, una per les mesures dels diàmetres i una pels nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix index de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms), i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà és inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considerereu que els valors d'entrada són positius (no cal fer la comprovació).

$$25 \times 5 = 125$$

$$4 + 4 = \boxed{8}$$

Exercici 2 (1 punt)
Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de pomes i les guardara en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardarà a la posició indicada de les taules corresponents.

En els dos casos, si s'introduix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anterior, es calcularà el nivell de qualitat del lot (utilitzar la funció QualitatLot de l'ex.1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$7 \times 7 = 49$$

$$125 + 4 = 129$$

$$\text{RESUM} : 25$$

Exercici 3 (1 punt)

Fer un procediment anomenat DistribucioPerNivells que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardara el numero de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardara el número de lots de cada nivell (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Nota: Si la taula no té la mateixa longitud que els nivells, el procediment no es executarà.

Exercici 5 (1 punt)

Fer una funció anomenada Estadistica que calculi i retorna alguns estadístics (mitjana i variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variància dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retornarà 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula: $\text{variancia} = \frac{1}{N-1} \sum_i (x_i - \mu)^2$ 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.

Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h.

Exercici 4 (1 punt)

Fer un procediment anomenat ConsultaPerNivell que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introdueixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguin el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.6 cms").

Si s'introduceix un nivell de qualitat que no és vàlid (nivells vàlids: 1, 2, 3, 4), es mostrara el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

Exercici 6 (2,5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que teniu implementat un procediment definit com:

```
void inicialitzartaula (int taula, int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilieu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula de distribució.
5. Utilitzar el procediment Entrarbadelots de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introduixi una de les opcions.
7. Liegr l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistrubucioPerNiveus de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0,5 punts)

Dedicar un nou tipus de dades, `TLot`, com un registre amb els camps: `Ident_Lot` (valor enter), `Diametre` (valor real), `Qualitat` (valor enter) i `Origen_Lot` (cadena de 50 caràcters).

Fer un procediment anomenat `OmplirLot` que rebi com a paràmetre un registre del tipus `TLot` i omplir els seus camps a partir de la informació entrada per teclat.

Exercici 9 (1 punt)

Fer una funció anomenada `BuscarPrimerZero` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retorna -1.

Exercici 8 (1 punt)

Fer un procediment anomenat `QuadratMatrxiu` que elevi al quadrat tots els elements d'una matrxiu d'enters de n files i 5 columnes. El procediment rebria com a paràmetres: una matrxiu d'enters de 5 columnes, el número de files de la matrxiu, i el número de columnes de la matrxiu.

Nota: Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`

1533039

Float Comprobacio Prestec (Quant Prestec, Float salari anual, Float Valor propietat)

{
 Float Total;

Total = 0;

 IF (salari anual \geq Prestec) || salari anual == (Prestec)

Total = Total + 5;

else:

 IF (salari anual \geq $\frac{Prestec}{4}$) || salari anual == ($\frac{Prestec}{4}$)

Total = Total + 3;

else:

 IF (salari anual \geq $\frac{Prestec}{10}$)

Total = Total + 1;

else:

IF (

fundamentos de Informática (103806) Curso 2019-2020
Examen Parcial (11 de Noviembre de 2019) **Grupo:** _____
Nombre estudiante: W. V. AM RAKSKA **NIU:** 9496777

Importante: Recuerde que hay que dar las mejores soluciones posibles en cada ejercicio. Además de funcionar correctamente, el código debe estar bien programado (código claro, con las instrucciones más adecuadas, sin operaciones ni variables innecesarias, etc.)

Las funciones y procedimientos de las preguntas de la 1 a la 6 forman parte de un único programa y, por tanto, debe haber coherencia entre sus definiciones, y la forma en que se utilizan en otras preguntas. El contexto del programa a desarrollar, lo tenéis a continuación:

AgroFruits, una cooperativa agrícola, quiere almacenar datos y hacer estadísticas sobre su producción de manzanas para mejorar la calidad de su producción. La producción de manzanas se distribuye a partir de lotes que vienen identificados por un número (entero), y para cada lote se quiere guardar el diámetro medio de las manzanas del lote y su nivel de calidad según un código numérico (1, 2, 3, 4).

Esta información se guardará en tres tablas (arrays), una para los identificadores, una para las medidas de los diámetros y una para los niveles de calidad). La información de un mismo lote se guarda en la misma posición de las tres tablas, es decir, el identificador de un lote, su diámetro medio y su nivel de calidad se guardan en el mismo índice de las tablas correspondientes.

En total, la cooperativa quiere guardar los datos de 100 lotes de manzanas.

Ejercicio 1 (1 punto)

Hacer una función llamada CalidadLote para clasificar un lote en uno de los niveles de calidad marcados (1, 2, 3 ó 4). La función recibirá el valor del diámetro medio del lote (en cms) y devolverá el nivel numérico de calidad del lote según el siguiente criterio:

- nivel 1: si el diámetro medio es inferior a 5 cms.
- nivel 2: si el diámetro medio es superior o igual a 5 cms e inferior a 7 cms.
- nivel 3: si el diámetro medio es superior o igual a 7 cm e inferior a 9 cms.
- nivel 4: si el diámetro medio es superior o igual a 9 cms.

NOTA: Considere que los valores de entrada son positivos (no hay que hacer la comprobación).

$$395 - 55 = \underline{\underline{340}}$$
$$340 + 247 = \underline{\underline{587}}$$

$$168 \times 4 = 672$$

$$\text{Resultat} : 672$$

Lo que se pide es una función EntrarDatosLote que pedirá los datos de un lote de manzanas y los guardará en las tablas correspondientes. La función recibirá como parámetros:

- las tres tablas con los identificadores, los diámetros medios y los niveles de calidad
- un entero con la dimensión de las tablas anteriores (un solo valor)
- la posición de la tabla donde se quieren guardar los datos que se introduzcan

La función comprobará que la posición en la que desea guardar los datos sea un índice correcto (no inferior a cero, ni superior al último índice válido de las tablas). Si la posición es incorrecta, la función devolverá -1 y no se hará nada más.

Si la posición es correcta, se pedirá al usuario que introduzca el valor del identificador del lote y el diámetro medio de las manzanas del lote, y se guardarán en la posición indicada de las tablas correspondientes.

En ambos casos, si se introduce un valor que no es positivo, se mostrará el mensaje "Error: Valor incorrecto" y se volverá a pedir el valor. Este proceso se repetirá hasta que se introduzcan los valores correctamente.

Una vez introducidas correctamente los datos anteriores, se calculará el nivel de calidad del lote (utilice la función CalidadLote del ej.) y se guardará en la tabla correspondiente.

Si se han introducido todos los datos correctamente, la función devolverá 0.

Ejercicio 3 (1 punto)

Hacer un procedimiento llamado `DistribucionPorNiveles` que calcule cuántos lotes hay de cada nivel de calidad y lo guarde en una tabla (array). El procedimiento recibirá como parámetros: una tabla con los niveles de calidad de los lotes, la dimensión de la tabla anterior (la pasada como primer parámetro), y una tabla donde se guardará el número de lotes de cada nivel de calidad en una posición diferente.

NOTA: La tabla donde se guardará el número de lotes de cada nivel (tercer parámetro) estará inicializada a 0 (no hay que inicializarla, en este procedimiento).

Ejercicio 4 (1 punto)

Hacer un procedimiento llamado `ConsultaPorNivel` que permita visualizar los datos de todos los lotes de un determinado nivel de calidad. El procedimiento recibirá como parámetros:

- las tres tablas con los identificadores, los diámetros medios y los niveles de calidad
- un entero con la dimensión de las tablas anteriores (ún solo valor)

El procedimiento pedirá al usuario que introduzca el nivel de calidad a consultar. A continuación, el procedimiento mostrará por pantalla los datos (identificador y diámetro medio) de los lotes que tengan el nivel de calidad indicado por el usuario. Los datos de un mismo lote se mostrarán en una misma línea (p.ej. "ID: 1234 diámetro: 5.6cm").

Si se introduce un nivel de calidad que no es válido (niveles válidos: 1, 2, 3, 4), se mostrará el mensaje "Error: Nivel de calidad inexistente" y se saldrá del procedimiento.

Vaik Iñaki y Carla (Gest. Gestión - Informática)

Ejercicio 5 (1 punto)

Hacer una función llamada `Estadistica` que calcule y devuelva algunos estadísticos (media y varianza) calculados sobre una tabla de valores reales. La función recibirá como parámetros:

- una tabla de valores reales
 - dos variables reales para guardar la media y la varianza de los valores de la tabla
- El valor de retorno de la función será -1 si la dimensión entrada como segundo parámetro es incorrecta (más pequeña que 1) y por tanto no se pueden calcular los estadísticos. Si la dimensión es correcta y se pueden calcular correctamente la media y la varianza, la función devolverá 0.

Nota: Para calcular la varianza de un conjunto de datos, debe seguir la fórmula:

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

donde x_i son los elementos del conjunto de datos, μ es la media de los datos y N es el número de elementos del conjunto de datos.

Para elevar un valor al cuadrado, se puede utilizar la función `pow (valor, 2)` de la librería `math.h`

Ejercicio 6 (25 puntos)
· Hacer un programa completo (declaraciones globales y función main()) que siga los siguientes pasos:

1. Declarar (donde corresponda) las constantes necesarias para que el programa sea fácilmente modificable.
2. Declarar (donde corresponda) tres estructuras de datos denominadas identificadores, medidas y niveles para guardar los identificadores de los lotes de manzanas a estudiar, los valores de sus diámetros medios y sus niveles de calidad, respectivamente.
3. Declarar (donde corresponda) una estructura de datos llamada distribucion para guardar el número de lotes que hay de cada nivel de calidad.
4. Suponed que tenéis implementado un procedimiento definido como:

```
void InicializarTabla ( int tabla[], int nElementos );
```

que permite inicializar a cero una tabla de enteros de tamaño nElementos. Utilizar (NO IMPLEMENTAR) este procedimiento para inicializar a cero la tabla distribucion.
5. Utilizar el procedimiento EntrarDatosLote del ej. 2 para pedir los datos de TODOS los lotes y guardarlas en las estructuras de datos identificadores, medidas y niveles.
6. Mostrar un mensaje indicando que tiene 4 opciones disponibles (1-Distribución por calidades, 2- Consulta de lotes por grupo, 3-Estadísticos, 4-Salir) y pedir al usuario que introduzca una de las opciones.
7. Leer la opción elegida.
8. Si la opción elegida es 1:
 - 8.1. Llamar al procedimiento DistribucionPorNiveles del ej. 3 para calcular cuántos lotes hay de cada nivel de calidad.
 - 8.2. Mostrar por pantalla, cuántos lotes hay de cada nivel de calidad con mensajes del estilo:
"Nivel 1: 12 lotes"
9. Si la opción elegida es 2, llamar al procedimiento ConsultaPorNivel del ej. 4 para visualizar por pantalla los datos de todos los lotes de un nivel de calidad concreto.
10. Si la opción elegida es 3:
 - 10.1. Llamar al procedimiento Estadistica del ej. 5 para calcular la media y la varianza de las medidas (diámetros) de los lotes de manzanas.
 - 10.2. Si la función devuelve un error, mostrar el mensaje "Error: No se han podido calcular los estadísticos". En caso contrario, mostrar por pantalla los valores de la media y la varianza.
11. Si la opción elegida no es ninguna de las 4 disponibles, mostrar el mensaje "Opcion incorrecta".
12. Repetir los pasos del 6 al 11, hasta que el usuario seleccione la opción de salir.

Ejercicio 7 (0.5 puntos)

Declarar un nuevo tipo de datos, **Tlote**, como un registro con los campos: **Ident_lote** (valor entero), **Diámetro** (valor real), **Calidad** (valor entero) y **Origen_lote** (cadena de 50 caracteres). Hacer un procedimiento llamado **LlenarLote** que reciba como parámetro un registro del tipo **Tlote** y llene sus campos a partir de la información entrada por teclado.

Ejercicio 9 (1 punto)

Hacer una función llamada **BuscarPrimerCero** que reciba como parámetros un vector de enteros y un valor entero con el número de elementos del vector. La función debe calcular y devolver en qué índice del vector esta el primer elemento igual a cero. Si el vector no contiene ningún cero, devolverá -1.

Ejercicio 8 (1 punto)

Hacer un procedimiento llamado **CuadradoMatriz** que eleve al cuadrado todos los elementos de una matriz de enteros de n filas y 5 columnas. El procedimiento recibirá como parámetros: una matriz de enteros de 5 columnas, el número de filas de la matriz, y el número de columnas de la matriz.

Nota: Para elevar un valor al cuadrado, se puede utilizar la función `pow(valor,2)` de la librería `math.h`.

include <iostreams>
define parameters vars
using namespace std;

define DTM
define DTW

int identifyclass(polygonset)
int minj
int maxj
int n;

Initializar Array (V[1], DTM, DTW);

Max Minpas();

Switch (option)

case 1: "introduce" & "denkprocedur" der parameteren für int vektor n & vektor m

~~the calculate function needs~~

while (darkprocedur & vektor n und vektor m != 0)

Vektor (max, max);

Initializar Array (V[1], DTM, DTW);

case 2: MaxMin Array (V[1])



Examen Parcial (11 de Novembre de 2019)

Nom estudiant: Joseph Clavilorne Simó

Curs 2019-2020

Grup: NIU: 4384911

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherentia entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenui a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot és voli guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una per als identificadors, una per les mesures del nombre de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu nombre mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Per una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$\begin{array}{r} 4 + 4 = 8 \\ 8 - 2 = \underline{\underline{6}} \end{array}$$

Exercici 2 (1 punt)

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- la posició de la taula on es volen guardar les dades que s'introduixin
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index valid de les taules). Si la posició és incorrecta, la funció retornara -1 i no es farà res més.

Sí la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardarán a la posició indicada de les taules corresponents.

En els dos casos, si s'introduix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzant la funció CodiVendaLot de l'ex.1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$5 + 20 = 25$$

$$25 \times 2 = 50$$

$$\text{Resultat} = 50$$

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodis` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada codi de venda en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorni alguns estadístics (màxim, mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodí` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el codi de venda que vol consultar. A continuació, el procediment mostrà per pantalla les dades (identificador i número mitja de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.e. "ID: 1234 num mitja de petals: 32.5").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codí de venda inexistent" i es sortirà.

Exercici 9 (2.5 punts)
Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula(int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introduixi una de les opcions.
7. Liegir l'opció escollida.
8. Si l'opció escollida és 1:
- 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
- 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodi de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
10. Si l'opció escollida és 3:
- 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
- 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.5 punts)

Declarar un nou tipus de clàdes, TIot, com un registre amb els camps: Ident_lot (valor enter), Num_mitja_petals (valor real), Codi_venda (valor enter) i Origen_lot (cadena de 50 caràcters).

Fer un procediment anomenat InicialitzarIot que rebi com a paràmetre un registre del tipus TIot i inicialitzi els seus camps amb valor zero als camps numèrics, i valor "...." al camp de tipus cadena de caràcters.

Exercici 9 (1 punt)

Fer una funció anomenada BuscarPrimerImparell que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat DobleMatriu que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el numero de columnes de la matriu.

int IndexMin(int arr[], int n) {

int i; int ans;

int valmin = arr[0];

int indexmin;

for (i=0; i < n; i++) {

if (arr[i] < valmin)

valmin = arr[i];

i = indexmin;

else if (arr[i] == valmin)

indexmin = i;

return indexmin;

else

if (salon > (pestec * $\frac{50}{100}$) and salon > (pestec * $\frac{25}{100}$)) {

pnts = pnts + 3;

else

if (salon > (pestec * $\frac{10}{100}$) and salon < (pestec * $\frac{25}{100}$)) {

pnts = pnts + 2;

else

if (salon > (pestec * 2)) {

pnts = pnts + 3;

else

if (salon > (pestec)) and salon < (pestec * 2) {

pnts = pnts + 3;

return pnts;

2-

```

int voter( int min, int max ) {
    int i;
    int vot;
}

```

cat < "Enter voter entre" & min & ":" & max & endl;

if (i > max) "Error voter not" & endl;

else if (i < min) "Error voter not" & endl;

for (i=0; i<parlementaires; i++) {

cin >> vot;

while (vot < min || vot > max) {

cout < "Error parlementaire vote de rang" & endl;

cin >> vot;

}

return 0;

return 1;

else

return 0;

4-

bad no zero array (int vt[], int DEM) {
int i;
for (i=0; i<DEM; i++) {

if (vt[i] == 0)

cout < "Error DEM" & endl;

cat < "Enter DEM entre" & DEM & ":" & endl;

if (DEM < min || DEM > max) "Error DEM not" & endl;

else if (DEM < min) "Error DEM not" & endl;

for (i=0; i<DEM; i++) {

cout < "Enter vt[" & i & "] " & endl;

cin >> vt[i];

}

void Escriture(char* vt[DEM], int DEM, int RST)

int i;

for (i=0; i<DEM; i++) {

cout < vt[i] << " " << endl;

}

cout < endl;

Resultats [vt[DEM - 1] + 1]

3

2

1

Exercicis d'Informativa (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: 9518211
Nom estudiant: William Gruta Nowotny NIU: 9518211

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot és vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix index de les taules corresponents. En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 o 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$100 + 40 = 140$$
$$200 - 20 = \boxed{180}$$

Exercici 2 (1 punt)

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardarán a la posició indicada de les taules corresponents.

En els dos casos, si s'introduix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament. Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzeu la funció CodiVendaLot de l'ex.1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$13 - 3 = 10$$

$$15 \times 3 = 45$$

$$\text{Resultat} = 45$$

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodi` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer parametre), i una taula on es guardarà el numero de lots de cada codi de venda en una posició different.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introdueixi el codi de venda que vol consultar. A continuació, el procediment mostrara per pantalla les dades (identificador i número mitja de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.6").

Si s'introduceix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codi de venda inexistent" i es sortirà.

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorna alguns estadístics (màxim, mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

Esercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi: 10: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodí de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0,5 punts)

Declarar un nou tipus de dades, `TLot`, com un registre amb els camps: `Ident_lot` (valor enter), `Num_mitja_petais` (valor real), `Codi_venda` (valor enter) i `Oxigen_lot` (cadena de 50 caràcters).

Fer un procediment anomenat `IniciarLot` que rebi com a paràmetre un registre del tipus `TLot` i inicialitzi els seus camps amb valor zero als camps numèrics, i valor "...." al camp del tipus cadena de caràcters.

Exercici 9 (1 punt)

Fer una funció anomenada `BuscarPrimerImparell` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat `DobleMatriu` que doblí el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

$\text{if}(\text{operand1} + \text{operand2} > 0) \& \& (x_1 != x_2) \& \& (y_1 != y_2)$

196586

operand1 + operand2 > 0
operator1 + operator2 > 0

bool ignals = false;

IF ($x_1 == x_2 \& y_1 == y_2$)

T2

$x_1 != x_2$
 $y_1 != y_2$

ignals = true

$$\text{sol bri} = 1000 \epsilon$$

$$\underline{\text{pres tec}} = 500 \epsilon$$

$$0.5 = 250$$

$[2, 3, 4]$

a
b
c

$[0, 0, 1, 1, 0]$

$\begin{pmatrix} 1 & 2 \\ 2 & 3 \end{pmatrix}$



Exercicis d'informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: 7064134

Nom estudiant: Joseph Smith Graham NIU:

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenui a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter) i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una per als identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

Exercici 1 (1 punt)

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$30 \times 2 = \underline{30}$$

Exercici 2 (1 punt)
Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introduceix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demandar el valor. Aquest procés es repetirà fins que s'introdueixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzu la funció CodiVendaLot de l'ex.1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$12 + 12 = 24$$

$$23 + 2 = 25$$

Resultado : 5

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodi` que calculi quants lòts hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lòts, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lòts de cada codi de venda en una posició diferent.

NOTA: La taula on es guardarà el número de lòts de cada codi de venda (tercer paràmetre) està inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lòts amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introdueixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i número mitja de pètals) dels lòts que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.6").

Si s'introduceix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrerà el missatge "Error: Codi de venda inexistent" i es sortirà.

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorna alguns estadístics (màxim, mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

Esercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
 2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
 3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.
 4. Suposeu que teniu implementat un procediment definit com:
- ```
void InicialitzarTaula(int taula[], int nElements);
```
- que permet inicialitzar a zero una taula d'enters de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
  6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
  7. Llegir l'opció escollida.
  8. Si l'opció escollida és 1:
    - 8.1. Cridar al procediment DistribuicioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
    - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"  9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodi de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
  10. Si l'opció escollida és 3:
    - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
    - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
  11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
  12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

**Exercici 7 (0,5 punts)**  
Declarar un nou tipus de dades, `TLot`, com un registre amb els camps: `Ident_lot` (valor enter), `Num_mitja_petals` (valor real), `Codi_venda` (valor enter) i `Origen_lot` (cadena de 50 caràcters).

Fer un procediment anomenat `IniciarLot` que rebi com a paràmetre un registre del tipus `TLot` i inicialitzi els seus camps amb valor zero als camps numèrics, i valor `"..."` al camp del tipus cadena de caràcters.

**Exercici 9 (1 punt)**

Fer una funció anomenada `BuscarPrimerImparell` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà `-1`.

**Exercici 8 (1 punt)**  
Fer un procediment anomenat `DobleMatriu` que doblí el valor de tots els elements d'una matriu d'enters de  $n$  files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

case 3 i : if ( $b = \ell$ )

{

$c1 = \text{IndexMinimNoZerWay}(\text{Results}, N\_DIPUTATS);$

$c2 = \text{IndexMaximWay}(\text{Results}, N\_DIPUTATS);$

cout << "La Fubada es el num." << c1 << "\n"  
la selliscada es el num." << c2 ;

else

cout << "Error en la imprenta de " j

break;

case 4 : // Es el de sortir

break;

default : cout << "Opcio no posseu" j

break;

} while ( opcio != 4 );

return 0;

Struct TParlament {

~~No~~ char Nom[10];

char Cognoms[20];

~~No~~ int Vot;

} j

int main ()

{

TParlament A;

A.Nom = "A. Nom";

A.Cognoms = "A. Cognoms";

A.Vot = 0;

} j

## Int NormalVector

```
#include <math.h>
```

```
Int NormalVector (x1, y1, x2, y2, &result),
```

```
{
 bool bj; float totalx; float? sum);
```

```
 b = ((x2 - x1) & B (y2 - y1))
```

```
 if (b == 0)
```

```
 return !bj
```

```
 else {
```

```
 totalx = (x2 - x1)
```

```
 totalx = (y2 - y1)
```

```
 totalx = pow (totalx, 2);
```

```
 totalx = pow (totalx, 2);
```

```
 sum = float d * float z;
```

```
 result = sqrt (sum);
```

```
 }
 return !bj
}
```

```
}
```

**Examen Parcial (11 de Novembre de 2019)**

**Grup:**  
**NIU: 2154928**

**Nom estudiant:** DÍNEA VÍDEO GARCÍA

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherentia entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

AgroFruit, una cooperativa agrícola, vol enmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot és vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardará en tres taules (arrays) (una pels identificadors, una per les mesures dels diàmetres i una pels nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix index de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

**Exercici 1 (1 punt)**

Feu una funció anomenada `QualitatLot` per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà és inferior a 5 cms.
- nivell 2: si el diàmetre mitjà es superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$6 + 6 = \underline{12}$$

$$7 \times 7 = \underline{49}$$

$$20 + 20 = 40$$

$$36 + 3 = 39$$

$$\text{RESULTAT} = 79$$

Feu una funció anomenada `EntrarLot` que demanara les dades d'un lot de pomes i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introdueixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardaràn a la posició indicada de les taules corresponents.

En els dos casos, si s'introdueix un valor que no és positiu, es mostrerà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anterior, es calcularà el nivell de qualitat del lot (utilitzant la funció `QualitatLot` de l'ex.1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

Exercici 3 (1 punt)  
Fer un procediment anomenat DistribucioPerNivells que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardara el numero de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) estarà inicializada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)  
Fer una funció anomenada Estadística que calculi i retorni alguns estadístics (mitjana, variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula amb els valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals a guardar la mitjana i la variància dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retornarà 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:

$$\text{variància} = \frac{1}{N-1} \sum_i (x_i - \mu)^2 \quad \text{on } x_i \text{ són els elements del conjunt de dades, } \mu \text{ és la mitjana de les dades i } N \text{ és el nombre d'elements del conjunt de dades.}$$

Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

#### Exercici 4 (1 punt)

Fer un procediment anomenat ConsultaPerNivell que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguin el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 diametre: 5.6 cms").

Si s'introdueix un nivell de qualitat que no és valid (nivells valids: 1, 2, 3, 4), es mostrarà el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

Per un programa complet (declaracions globals i funció main) que segueixi els següents passus.

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
  2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
  3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
  4. Suposeu que teniu implementat un procediment definit com:
- ```
void InicialitzarTaula(int taula, int nElements);
```
- que permet inicialitzar a zero una taula d'enters de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment EntrarNadadesLot de l'ex. 2, per a demanar les clades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
 6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introduixi una de les opcions.
 7. Llegir l'opció escollida.
 8. Si l'opció escollida és 1:
 - 9.1. Cridar al procediment DistribucioPerNivells de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
 - 9.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 lots"
 9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
 10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
 11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
 12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.5 punts)
Declarar un nou tipus de dades, TIOT, com un registre amb els camps: Ident, lot (valor enter), Diametre (valor real), Qualitat (valor enter) i Origen_Lot (cadena de 50 caràcters).

Exercici 8 (1 punt)
Fer un procediment anomenat QuadratMatriu que elevi al quadrat tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

Exercici 9 (1 punt)
Fer un procediment anomenat AmpliarLot que rebi com a paràmetre un registre del tipus TIOT i ompli els seus camps a partir de la informació entrada per teclat.

Exercici 9 (1 punt)
Fer una funció anomenada BuscarPrimerZero que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retornar -1.

void ~~sort~~ Selection (vector<int> &arr, int resultset[])

int c
int b

while ((c < arr.size()) && (b == 0))

for (i = 0; i < arr[c].size(); i++)

resultset[i] = arr[c][i];

b = (arr[c][i] == 0);

}

if (b == 1)

arr[c].push_back(i);

else if (b == 0)

arr[c].push_back(i);

cout << "Index " << i << " : " << resultset[i] << endl;

else if (c == 0, pos == 0)

white(c, pos);

{ min = 0;

aux = 0;

g aux < 0;

for (i = 0; i < dig - 1; i++)

{ if ((dig - 1 - i) == 0) swap(aux, vec[i]);

else if (vec[i] > aux)

swap(vec[i], aux);

cout << "pos " << pos << endl;

28684

```
#include <conio.h>
```

```
using namespace std;
```

```
#define N_DIPUTATS 135
```

```
int main ()
```

```
{
```

```
int Vots [N_DIPUTATS], Resultats [N_DIPUTATS];
```

```
int opcio, v1, v2, p, i, c1, c2;
```

```
Initializar Array (Vots, N_DIPUTATS, 0);
```

```
MENU Principal ();
```

```
do {
```

```
cin >> opcio;
```

```
switch (opcio)
```

```
{
```

```
Case 1 : cin >> v1;
```

```
while e ((v1 > N_DIPUTATS) || (v1 < 0)) {
```

```
    cin >> v1;
```

```
}
```

```
v2 = Votar (-o, N_DIPUTATS);
```

```
Vots [v1 - 1] = v2;
```

```
break;
```

```
Case 2 : p = NoZeroArray (Vots, N_DIPUTATS);
```

```
if (p == 1)
```

```
{
```

```
    Inicialitzar Array (Resultats, N_DIPUTATS, 0);
```

```
    Estadisticas (Vots, N_DIPUTATS, Resultats);
```

```
    for (i = 0; i < N_DIPUTATS; i++)
```

```
        cout << "Els vots obligatius pel parlamentari en la fila " << i << " es " << c
```

```
        Resultats[i];
```

```
}
```

```
    else cout << "Encara no s'ha fet l'estadistica";
```

```
    break;
```

Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: 1-A

Nom estudiant: Donald Hesley Baxter

NIU: 5660994

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions. I la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

AgroFruits, una cooperativa agrícola, vol enmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4). Aquesta informació es guardará en tres taules (arrays) (una per als identificadors, una per les mesures dels diàmetres i una per els nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix índex de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot, per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà es inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$30 \times 10 = 300$$
$$300 + 50 = \underline{\underline{350}}$$

$$4 \times 8 = 32$$

$$32 \times 2 = 64$$

Resultat : 64

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de pomes i les guardará en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introdueixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardarán a la posició indicada de les taules corresponents.

En els dos casos, si s'introduïx un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anterioris, es calcularà el nivell de qualitat del lot (utilitzeu la funció QualitatLot de l'ex.1) i es guardará a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerNivells` que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerNivel` que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrà per pantalla les dades (identificador i diàmetre mitjà) dels lots que tingui el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostren en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.6 cms").

Si s'introdueix un nivell de qualitat que no és vàlid (nivells vàlids: 1, 2, 3, 4), es mostrarà el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

Exercici 5 (1 punt)

Fer una funció anomenada `Estatistica` que calculi i retorni alguns estadístics (mitjana i variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variància dels valors de la taula (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és incorrecta poden calcular correctament la mitjana i la variància, la funció retornarà 0.

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retornarà 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:

$$\text{variancia} = \frac{1}{N-1} \sum_i (x_i - \mu)^2 \quad \text{on } x_i \text{ són els elements del conjunt de dades, } \mu \text{ és la mitjana de les dades i } N \text{ és el nombre d'elements del conjunt de dades.}$$

Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`.

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que teniu implementat un procediment definit com:


```
void InicialitzarTaula (int taula, int nElements);
```

 que permet inicialitzar a zero una taula d'enters de mida nElements. Utilitzau (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.
5. Utilitzar el procediment EntradaDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introduixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerNivells de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 Lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estandistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.5 punts)
Declarar un nou tipus de dades, TILOT, com un registre amb els camps: Ident_lot (valor enter),
diàmetre (valor real), Qualitat (valor enter), Origen_lot (cadena de 50 caràcters).

Fer un procediment anomenat OmplirLilot que rebi com a paràmetre un registre del tipus TILOT
i ompli els seus camps a partir de la informació entrada per teclat.

Exercici 9 (1 punt)
Fer una funció anomenada BuscarPrimerZero que rebi com a paràmetres un vector
d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar
en quin índex del vector hi ha el primer element igual a zero. Si el vector no conté cap zero,
retornarà -1.

Exercici 8 (1 punt)

Fer un procediment anomenat QuadratMatriu que elevi al quadrat tots els elements d'una
matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu
d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

Note: Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

1526394

```

int maxos in 0
{
    int Vots [N-Digitats];
    int Resultats [N-Digitats];
    int aux = 0;
    int i;
    int min;
    int max;

    int antic;
    int numero;
    int cont;

    InicialitzarArray (Vots, 0, N-Digitats, 0);
    InicialitzarArray (Resultats, 0, N-Digitats, 0);

    do
        menuPrincipal ();
        cin >> antic;
        switch (antic)
        {
            case 1:
                do
                    {
                        cin >> numero;
                    } while (numero > N-Digitats || numero < 1);
                    Vots [numero - 1] = votar (1, N-Digitats);
                    break;
                case 2:
                    aux = NuZeroArray (Vots, N-Digitats);
                    if (aux == 1)
                    {
                        InicialitzarArray (Resultats, N-Digitats, 0);
                        Escriutini (Vots, resultado, N-Digitats, Resultats);
                        for (i = 0; i < N-Digitats; i++)
                            cout << Resultats [i] << endl;
                    }
                    else
                        cout << "Envia més es post. per l'escriutini" << endl;
                    break;
        }
    }
}

```

while (true) {
 cout << "Enter the number to calculate: ";
 cin >> num;
 cout << "The factorial of " << num << " is "
 cout << fact(num);
}

int fact(int n) {
 if (n == 0) return 1;
 else return n * fact(n - 1);
}

Fornaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: 43

Nom estudiant: Miquel Hernández NIU: 8572783

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

AgroFruits, una cooperativa agrícola, vol emmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardarà en tres taules (arrays) (una pels identificadors, una per les mesures dels diàmetres i una pels nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix índex de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot, per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà és inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

Exercici 2 (1 punt)
Fer una funció anomenada EntrarDadesLot, que demanarà les dades d'un lot de pomes i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició es correcta, es demanarà a l'usuari que introdueixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardarán a la posició indicada de les taules corresponents.

En els dos casos, si s'introduïx un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anterior, es calcularà el nivell de qualitat del lot (utilitzar la funció QualitatLot de l'ex.1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$4 \times 3 = 12$$

$$12 \times 2 = 24$$

Resol (1): 24

$$\begin{array}{r} 20 \times 10 = 200 \\ 200 + 50 = \underline{\underline{250}} \end{array}$$

Exercici 3 (1 punt)

Fer un procediment anomenat DistribucióPerNivells que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) està inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 5 (1 punt)

Fer una funció anomenada Estadística que calculi i torni alguns estadístics (mitjana i variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variància dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retornarà 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:

$$\text{variancia} = \frac{1}{N-1} \sum_i (x_i - \mu)^2 \quad \text{on } x_i \text{ són els elements del conjunt de dades, } \mu \text{ és la mitjana de les dades i } N \text{ és el nombre d'elements del conjunt de dades.}$$

Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

Exercici 4 (1 punt)

Fer un procediment anomenat ConsultaPerNivel que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrara per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguin el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.6 cms").

Si s'introdueix un nivell de qualitat que no és vàlid (nivells vàlids: 1, 2, 3, 4), es mostrarà el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

Exercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que teríeu implementat un procediment definit com:

```
void Inicialitzaraula (int traula, int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.
5. Utilitzar el procediment EntradaLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles ('1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir') i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
 - 8.1. Cridar al procediment DistribucioPerNiveus de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
 - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil:
"Nivell 1: 12 lots."
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
 - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
 - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.5 punts)

Declarar un nou tipus de dades, TIot, com un registre amb els camps: Ident_Lot (valor enter), Diàmetre (valor real), Qualitat (valor enter) i Origem_Lot (cadenes de 50 caràcters). Fer un procediment anomenat AmpliarLot que rebi com a paràmetre un registre del tipus TIot i ompli els seus camps a partir de la informació entrada per teclat.

Exercici 8 (1 punt)

Fer un procediment anomenat QuadratMatriu que elevi al quadrat tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el numero de files de la matriu, i el numero de columnes de la matriu.
Nota: Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

Exercici 9 (1 punt)

Fer una funció anomenada BuscarPrimerZero que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retorna -1.

Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: 4T

NIU: 9151233

Nome: Júlia Dejuvis Kelly

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherentia entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

AgroFruits, una cooperativa agrícola, vol enmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot és vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardarà en tres taules (arrays) (una per als identificadors, una per les mesures dels diàmetres i una per als nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix index de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà és inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$200 \times 10 = 2000$$
$$200 + 30 = \underline{\underline{230}}$$

Exercici 2 (1 punt)

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de pomes i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex valid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introdueixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardaran a la posició indicada de les taules corresponents.

En els dos casos, si s'introdueix un valor que no és positiu, es mostrara el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anterior, es calcularà el nivell de qualitat del lot (utilitzant la funció QualitatLot de l'ex.1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerNivells` que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerNivell` que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variancia dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variancia, la funció retornarà 0.

Nota: Per a calcular la variancia d'un conjunt de dades, heu de seguir la fórmula:

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

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

Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`.

Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorna alguns estadístics (mitjana i variancia) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variancia dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variancia, la funció retornarà 0.

Nota: Per a calcular la variancia d'un conjunt de dades, heu de seguir la fórmula:

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

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

Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`.

Exercici 6 (2.5 punts)

Fer un programa complet (declaracions globals i funció main()) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
 2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
 3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
 4. Suposeu que teniu implementat un procediment definit com:
- ```
Void InicialitzarTaula (int taula, int nElements);
```
- que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades i identificadors, mesures i nivells.
  6. Mostrar un missatge indicant ' té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estatístics, 4-Sortir) i demanar a l'usuari que introduixi una de les opcions.
  7. Llegir l'opció escollida.
  8. Si l'opció escollida és 1:
- 8.1. Cridar al procediment DistribucioPerNivells de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
  - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
  10. Si l'opció escollida és 3:
- 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
  - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
  12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

**Exercici 7 (0.5 punts)**

Declarar un nou tipus de dades, TLot, com un registre amb els camps: Ident\_Lot (valor enter), Diametre (valor real), Qualitat (valor enter) i Oxiogen\_Lot (cadena de 50 caràcters). Fer un procediment anomenat AmpliarLot que rebi com a paràmetre un registre del tipus TLot i ompli els seus camps a partir de la informació entrada per teclat.

**Exercici 9 (1 punt)**

Fer una funció anomenada BuscarPrimerZero que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retornarà -1.

**Exercici 8 (1 punt)**

Fer un procediment anomenat QuadratMatriu que elevi al quadrat tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu i el número de columnes de la matriu.

Nota: Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

# Fonaments d'informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: 47

Nom estudiant: WILHELM KÜBLER RODRÍGUEZ NIU: 2119657

**Important:** Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació.

AeroFruits, una cooperativa agrícola, vol emmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot és vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardará en tres taules (arrays) (una pels identificadors, una per les mesures dels diàmetres i una pels nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix index de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

## Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà es inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$\begin{array}{r} 20 \times 10 = 200 \\ 200 + 50 = \underline{\underline{250}} \end{array}$$

$$4 \times 3 = 12 \quad 12 \times 2 = 36$$

$$\text{Resol: } 4 \times 36$$

**Exercici 2 (1 punt)**  
Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de pomes i les guarderà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introduixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardaran a la posició indicada de les taules corresponents.

En els dos casos, si s'introdueix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el nivell de qualitat del lot (utilitzant la funció QualitatLot de l'ex.1) i es guardara a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

### Exercici 3 (1 punt)

Fer un procediment anomenat DistribucioPerNivells que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

### Exercici 5 (1 punt)

Fer una funció anomenada Estadistica que calculi i retorna alguns estadístics (mitjana i variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variància dels valors de la taula

El valor de retorn de la funció sera. -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament la mitjana i la variància, la funció retornarà 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:

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

on  $x_i$  són els elements del conjunt de dades,  $\mu$  és la mitjana de les dades i  $N$  és el nombre de elements del conjunt de dades.

Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

### Exercici 4 (1 punt)

Fer un procediment anomenat ConsultaPerNivell que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduís el nivell de qualitat que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguin el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.6 cms").

Si s'introdueix un nivell de qualitat que no és valid (nivells vàlids: 1, 2, 3, 4), es mostrerà el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

**Exercici 6 (25 punts)**

Fer un programa complet (declaracions globals i funció main() ) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula, int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements; IMPLEMENTEU aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introduixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
  - 8.1. Cridar al procediment DistribucioPerNivells de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
  - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
  - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
  - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

**Exercici 7 (0,5 punts)**

Declarar un nou tipus de dades, TIot, com un registre amb els camps: Ident\_lot (valor enter), Diametre (valor real), Qualitat (valor enter) i Origen\_lot (cadena de 50 caràcters). Fer un procediment anomenat OmplirIOT que rebi com a paràmetre un registre del tipus TIOT i ompli els seus camps a partir de la informació entrada per teclat.

**Exercici 9 (1 punt)**

Fer una funció anomenada BuscarPrimerZero que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retornar -1.

**Exercici 8 (1 punt)**

Fer un procediment anomenat QuadraMatriu que elevi al quadrat tots els elements d'una matrui d'enters de n files i 5 columnes. El procediment rebra com a paràmetres: una matrui d'enters de 5 columnes, el número de files de la matrui, i el número de columnes de la matrui.

Nota: Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

#include <iostream>

2533543

#define DIM 135

int main()

{ int opcion, vot=0, votar, comp  
int votos[DIM], resultados[DIM];

Inicializar Array (votos, DIM, 0);

MenuPrincipal();

cin >> opcion;

switch (opcion)

{

case 1: while ((vot < 1) || (vot > 135)) {

cin >> vot

{  
Votar = Votar (1, 135);

Votos [vot] = votar;

break;

case 2: comp = NoZeroArray (votos, DIM)

{ if (comp == 0)

cout << "Encara no es pot fer l'escrutini";

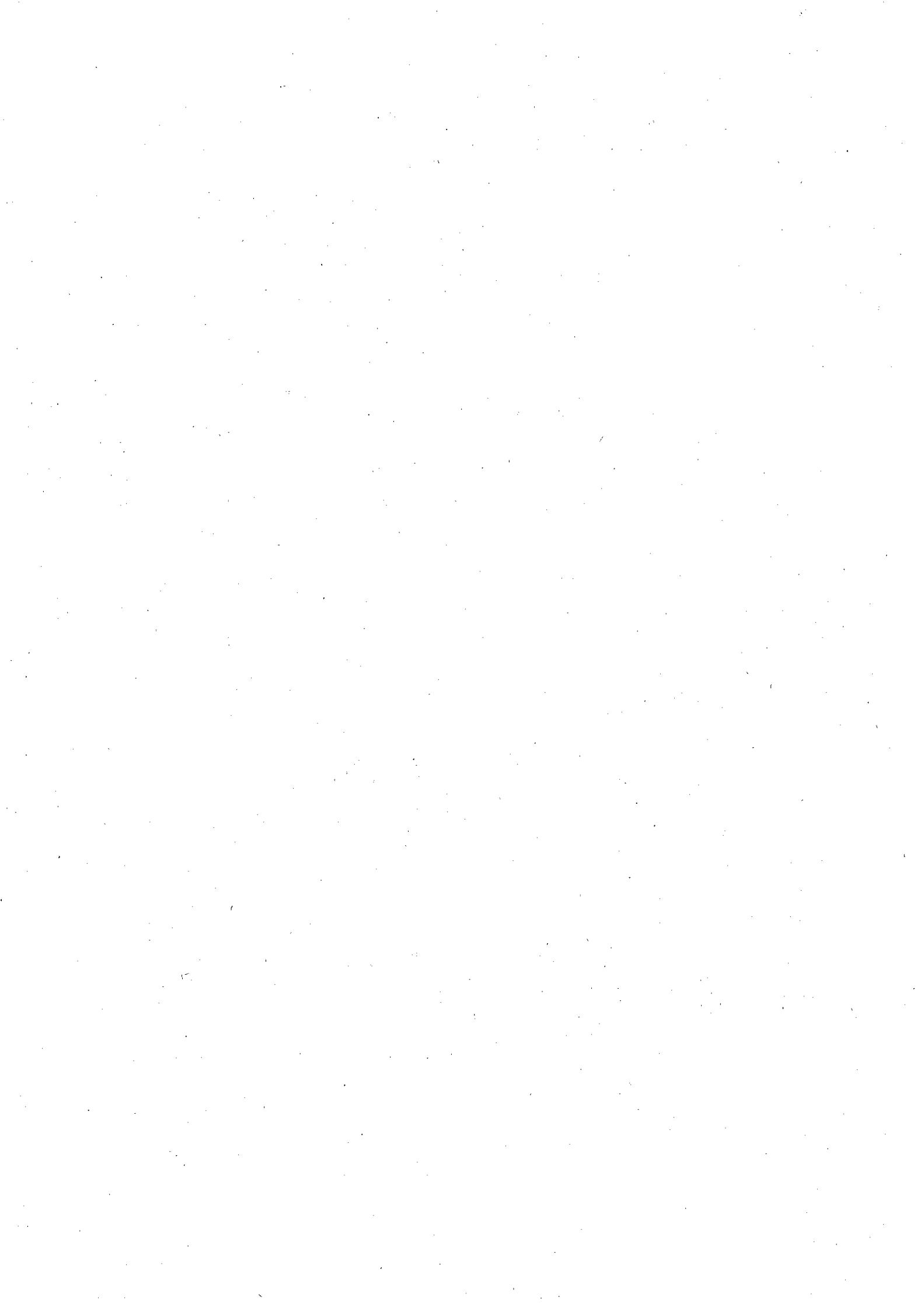
else

Inicializar Array (resultados, DIM, 0);

{ if (k == 0)

cout << "Encara no hi ha censutini";

else



# Fonaments d'Informàtica (103806)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Grup: 5741750

Nom estudiant: Henry Lovelace Price

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenui a continuació:

AgroFruit, una cooperativa agrícola, vol emmagatzemar dades i fer estadístiques sobre la seva producció de pomes per tal de millorar la qualitat de la seva producció. La producció de pomes es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot és vol guardar el diàmetre mitjà de les pomes del lot i el seu nivell de qualitat segons un codi numèric (1,2,3,4).

Aquesta informació es guardará en tres taules (arrays) (una pels identificadors, una per les mesures dels diàmetres i una pels nivells de qualitat). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu diàmetre mitjà i el seu nivell de qualitat es guarden al mateix index de les taules corresponents.

En total, la cooperativa vol guardar les dades de 100 lots de pomes.

## Exercici 1 (1 punt)

Fer una funció anomenada QualitatLot per a classificar un lot en un dels nivells de qualitat marcats (1, 2, 3 ó 4). La funció rebrà el valor del diàmetre mitjà del lot (en cms) i retornarà el nivell numèric de qualitat del lot segons el criteri següent:

- nivell 1: si el diàmetre mitjà és inferior a 5 cms.
- nivell 2: si el diàmetre mitjà és superior o igual a 5 cms i inferior a 7 cms.
- nivell 3: si el diàmetre mitjà és superior o igual a 7 cms i inferior a 9 cms.
- nivell 4: si el diàmetre mitjà és superior o igual a 9 cms.

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$5 + 9 = 14$$

$$14 \times 27 = \underline{\underline{378}}$$

**Exercici 2 (1 punt)**  
Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de pomes i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introdueixi el valor de l'identificador del lot i el diàmetre mitjà de les pomes del lot, i es guardarán a la posició indicada de les taules corresponents.

En els dos casos, si s'introdueix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el nivell de qualitat del lot (utilitzar la funció QualitatLot de l'ex1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$11 + 11 = 22$$

$$22 - 1 = 20$$

Resultat: 20

### Exercici 3 (1 punt)

Fer un procediment anomenat DistribucioPerNivells que calculi quants lots hi ha de cada nivell de qualitat i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els nivells de qualitat dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada nivell de qualitat en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada nivell (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

### Exercici 4 (1 punt)

Fer un procediment anomenat ConsultaPerNivel que permeti visualitzar les dades de tots els lots d'un determinat nivell de qualitat. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els diàmetres mitjans i els nivells de qualitat
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanara a l'usuari que introduixi el nivell de qualitat que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i diàmetre mitjà) dels lots que tinguin el nivell de qualitat indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 diàmetre: 5.6 cms"). Si s'introdueix un nivell de qualitat que no és vàlid (nivells vàlids: 1, 2, 3, 4), es mostrarà el missatge "Error: Nivell de qualitat inexistent" i es sortirà.

### Exercici 5 (1 punt)

Fer una funció anomenada Estadistica que calculi i retorna alguns estadístics (mitjana i variància) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar la mitjana i la variància dels valors de la taula (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és incorrecta poden calcular correctament la mitjana i la variància, la funció retornarà 0.

Nota: Per a calcular la variància d'un conjunt de dades, heu de seguir la fórmula:  
$$\text{variancia} = \frac{1}{N-1} \sum_i (x_i - \mu)^2$$
 on  $x_i$  són els elements del conjunt de dades,  $\mu$  és la mitjana de les dades i  $N$  és el nombre d'elements del conjunt de dades.

Per a elevar un valor al quadrat, podeu utilitzar la funció pow(valor,2) de la llibreria math.h

**Exercici 6 (2,5 punts)**  
Fer un programa complet (declaracions globals i funció main() ) que segueixi els següents passos:

1. Declarar (on corresponguï) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on corresponguï) tres estructures de dades anomenades identificadors, mesures i nivells per a guardar els identificadors dels lots de pomes a estudiar, els valors dels seus diàmetres mitjans i els seus nivells de qualitat, respectivament.
3. Declarar (on corresponguï) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada nivell de qualitat.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula, int nElements);
```

que permet inicialitzar una taula d'enters de mida nElements utilitzant NO IMPLEMENTEU aquest procediment per a inicialitzar a zero la taula de distribució.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i nivells.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per qualitats, 2- Consulta de lots per grup, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
  - 8.1. Cridar al procediment DistribucioPerNivells de l'ex. 3 per a calcular quants lots hi ha de cada nivell de qualitat.
  - 8.2. Mostrar per pantalla, quants lots hi ha de cada nivell de qualitat amb missatges de l'estil: "Nivell 1: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerNivell de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un nivell de qualitat concret.
10. Si l'opció escollida és 3:
  - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular la mitjana i la variància de les mesures (diàmetres) dels lots de pomes.
  - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors de la mitjana i la variància.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

**Exercici 7 (0.5 punts)**

Declarar un nou tipus de dades, `Tlot`, com un registre amb els camps: `Ident_Lot` (valor enter), `Diametre` (valor real), `Qualitat` (valor enter) i `Origen_Lot` (cadena de 50 caràcters). Fer un procediment anomenat `OmplirLlot` que rebi com a paràmetre un registre del tipus `Tlot` i ompli els seus camps a partir de la informació entrada per teclat.

**Exercici 9 (1 punt)**

Fer una funció anomenada `BuscarPrimerZero` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element igual a zero. Si el vector no conté cap zero, retornarà -1.

**Exercici 8 (1 punt)**

Fer un procediment anomenat `QuadratMatriu` que elevi al quadrat tots els elements d'una matriu d'enters de  $n$  files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el numero de files de la matriu, i el numero de columnes de la matriu.

Note: Per a elevar un valor al quadrat, podeu utilitzar la funció `pow(valor,2)` de la llibreria `math.h`

int  $x \geq 0$ ,  $\text{char } x;$

0 1 2 3

while ( $x < \min(1, x_{\max})$ ) {  
  ~~do~~  
    out <= "

3 2 10

ResC

```
void Init(int a[], int Dim, int v) {
 int i;
 for (i=0; i<Dim; i++) {
 a[i]=v;
 }
}
```

```
void Gc_(int Vols[], Dim, ResC) {
 for (i=0; i<Dim; i++) {
 Vols[i] = ResC;
 }
}
```

Results[Vol[0]]++;

```
int Thr_(int a[], int Dim) {
 int i, j, min, max=a[0], min=0;
 for (i=0; i<Dim; i++) {
 if (a[i]> max) max = a[i];
 if (a[i]< min) min = a[i];
 }
 return min;
}
```

$$S \rightarrow S \otimes \gamma = 0.5q$$

$$3 \rightarrow S \geq -0.25q$$

$$1 \rightarrow S \geq 0.1q$$

$$S \rightarrow up \geq 2*q$$

$$3 \rightarrow up \geq 0.5q$$

# Fundamentos de Informática (103806) Curso 2019-2020

Examen Parcial (11 de Noviembre de 2019)

Grupo: 6890379

Nombre estudiante: Jenni López Martínez

Ejercicio 2 (1 punto)  
Hacer una función llamada EntrarDatosLote que pedirá los datos de un lote de manzanas y los guardará en las tablas correspondientes. La función recibirá como parámetros:

- las tres tablas con los identificadores, los diámetros medios y los niveles de calidad
- la posición de la tabla donde se quieren guardar los datos que se introduzcan
- un entero con la dimensión de las tablas anteriores (un solo valor)

Importante: Recuerde que hay que dar las mejores soluciones posibles en cada ejercicio. Además de funcionar correctamente, el código debe estar bien programado (código claro, con las instrucciones más adecuadas, sin operaciones ni variables innecesarias, etc.)

Las funciones y procedimientos de las preguntas de la 1 a la 6 forman parte de un único programa y, por tanto, debe haber coherencia entre sus definiciones, y la forma en que se utilizan en otras preguntas. El contexto del programa a desarrollar, lo tenéis a continuación:

AgroFruits, una cooperativa agrícola, quiere almacenar datos y hacer estadísticas sobre su producción de manzanas para mejorar la calidad de su producción. La producción de manzanas se distribuye a partir de lotes que vienen identificados por un número (entero), y para cada lote se quiere guardar el diámetro medio de las manzanas del lote y su nivel de calidad según un código numérico (1,2,3,4).

Esta información se guardará en tres tablas (arrays) (una para los identificadores, una para las medidas de los diámetros y una para los niveles de calidad). La información de un mismo lote se guarda en la misma posición de las tres tablas, es decir, el identificador de un lote, su diámetro medio y su nivel de calidad se guardan en el mismo índice de las tablas correspondientes.

En total, la cooperativa quiere guardar los datos de 100 lotes de manzanas.

## Ejercicio 1 (1 punto)

Hacer una función llamada CalidadLote para clasificar un lote en uno de los niveles de calidad marcados (1, 2, 3 ó 4). La función recibirá el valor del diámetro medio del lote (en cms) y devolverá el nivel numérico de calidad del lote según el siguiente criterio:

- nivel 1: si el diámetro medio es inferior a 5 cms.
- nivel 2: si el diámetro medio es superior o igual a 5 cms e inferior a 7 cms.
- nivel 3: si el diámetro medio es superior o igual a 7 cm e inferior a 9 cms.
- nivel 4: si el diámetro medio es superior o igual a 9 cms.

NOTA: Considere que los valores de entrada son positivos (no hay que hacer la comprobación).

$$450 - 50 = \underline{400}$$
$$400 - 200 = \underline{200}$$

$$105 \times 4 = 420$$

$$\text{Resultado} = 420$$

La función comprobará que la posición en la que desea guardar los datos sea un índice correcto (no inferior a cero, ni superior al último índice válido de las tablas). Si la posición es incorrecta, la función devolverá -1 y no se hará nada más.

Si la posición es correcta, se pedirá al usuario que introduzca el valor del identificador del lote y el diámetro medio de las manzanas del lote, y se guardarán en la posición indicada de las tablas correspondientes.

En ambos casos, si se introduce un valor que no es positivo, se mostrará el mensaje "Error: Valor incorrecto" y se volverá a pedir el valor. Este proceso se repetirá hasta que se introduzcan los valores correctamente.

Una vez introducidos correctamente los datos anteriores, se calculará el nivel de calidad del lote (utilice la función CalidadLote del ej.1) y se guardará en la tabla correspondiente.

Si se han introducido todos los datos correctamente, la función devolverá 0.

**Ejercicio 3 (1 punto)**

Hacer un procedimiento llamado `DistribucionPorNiveles` que calcule cuántos lotes hay de cada nivel de calidad y lo guarde en una tabla (array). El procedimiento recibirá como parámetros: una tabla con los niveles de calidad de los lotes, la dimensión de la tabla anterior (la pasada como primer parámetro), y una tabla donde se guardará el número de lotes de cada nivel de calidad en una posición diferente.

NOTA: La tabla donde se guardara el número de lotes de cada nivel (tercer parámetro) estará inicializada a 0 (no hay que inicializarla, en este procedimiento).

**Ejercicio 5 (1 punto)**

Hacer una función llamada `Estadistica` que calcule y devuelva algunos estadísticos (media y varianza) calculados sobre una tabla de valores reales. La función recibirá como parámetros:

- una tabla de valores reales
- un entero con la dimensión de la tabla anterior (la tabla pasada como primer parámetro)
- dos variables reales para guardar la media y la varianza de los valores de la tabla

El valor de retorno de la función será -1 si la dimensión entrada como segundo parámetro es incorrecta (más pequeña que 1) y por tanto no se pueden calcular los estadísticos. Si la dimensión es correcta y se pueden calcular correctamente la media y la varianza, la función devolverá 0.

Nota: Para calcular la varianza de un conjunto de datos, debe seguir la fórmula:

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

donde  $x_i$  son los elementos del conjunto de datos,  $\mu$  es la media de los datos y  $N$  es el número de elementos del conjunto de datos.

Para elevar un valor al cuadrado, se puede utilizar la función `pow (valor, 2)` de la librería `math.h`

**Ejercicio 4 (1 punto)**

Hacer un procedimiento llamado `ConsultaPorNivel` que permita visualizar los datos de todos los lotes de un determinado nivel de calidad. El procedimiento recibirá como parámetros:

- las tres tablas con los identificadores, los diámetros medios y los niveles de calidad
- un entero con la dimensión de las tablas anteriores (un solo valor)

El procedimiento pedirá al usuario que introduzca el nivel de calidad a consultar. A continuación, el procedimiento mostrará por pantalla los datos (identificador y diámetro medio) de los lotes que tengan el nivel de calidad indicado por el usuario. Los datos de un mismo lote se mostrarán en una misma línea (p.ej. "ID: 1234 diámetro: 5.6cm"). Si se introduce un nivel de calidad que no es válido (niveles válidos: 1, 2, 3, 4), se mostrará el mensaje "Error: Nivel de calidad inexistente" y se saldrá del procedimiento.

### Ejercicio 6 (2.5 puntos)

Hacer un programa completo (declaraciones globales y función main() ) que siga los siguientes pasos:

1. Declarar (donde corresponda) las constantes necesarias para que el programa sea fácilmente modificable.
2. Declarar (donde corresponda) tres estructuras de datos denominadas identificadores, medidas y niveles para guardar los identificadores de los lotes de manzanas a estudiar, los valores de sus diámetros medios y sus niveles de calidad, respectivamente.
3. Declarar (donde corresponda) una estructura de datos llamada distribucion para guardar el número de lotes que hay de cada nivel de calidad.
4. Suponed que tenéis implementado un procedimiento definido como:

```
void InicializarTabla (int tabla, int nElementos);
```

que permite inicializar a cero una tabla de enteros de tamaño nElementos. Utilizar (NO IMPLEMENTAR) este procedimiento para inicializar a cero la tabla distribucion.
5. Utilizar el procedimiento EntradaDatosLote del ej. 2, para pedir los datos de TODOS los lotes y guardarlas en las estructuras de datos identificadores, medidas y niveles.
6. Mostrar un mensaje indicando que tiene 4 opciones disponibles (1-Distribución por calidades, 2- Consulta de lotes por grupo, 3-Estadísticos, 4-Salir) y pedir al usuario que introduzca una de las opciones.
7. Leer la opción elegida.
8. Si la opción elegida es 1:
  - 8.1. Llamar al procedimiento DistribucionPorNiveles del ej. 3 para calcular cuántos lotes hay de cada nivel de calidad.
  - 8.2. Mostrar por pantalla, cuántos lotes hay de cada nivel de calidad con mensajes del estilo:  
"Nivel 1: 12 lotes"
9. Si la opción elegida es 2, llamar al procedimiento ConsultaPorNivel del ej. 4 para visualizar por pantalla los datos de todos los lotes de un nivel de calidad concreto.
10. Si la opción elegida es 3:
  - 10.1. Llamar al procedimiento Estadistica del ej. 5 para calcular la media y la varianza de las medidas (diámetros) de los lotes de manzanas.
  - 10.2. Si la función devuelve un error, mostrar el mensaje "Error: No se han podido calcular los estadísticos". En caso contrario, mostrar por pantalla los valores de la media y la varianza.
11. Si la opción elegida no es ninguna de las 4 disponibles, mostrar el mensaje "Opcion incorrecta".
12. Repetir los pasos del 6 al 11, hasta que el usuario seleccione la opción de salir.

**Ejercicio 7 (0.5 puntos)**

Declarar un nuevo tipo de datos, `TLote`, como un registro con los campos: `Ident_lote` (valor entero), `Diámetro` (valor real), `Calidad` (valor entero) y `Origen_lote` (cadena de 50 caracteres). Hacer un procedimiento llamado `LlenarLote` que reciba como parámetro un registro del tipo `TLote` y llene sus campos a partir de la información entrada por teclado.

**Ejercicio 9 (1 punto)**

Hacer una función llamada `BuscarPrimerCero` que reciba como parámetros un vector de enteros y un valor entero con el número de elementos del vector. La función debe calcular y devolver en qué índice del vector está el primer elemento igual a cero. Si el vector no contiene ningún cero, devolverá -1.

**Ejercicio 8 (1 punto)**

Hacer un procedimiento llamado `CuadradoMatriz` que eleve al cuadrado todos los elementos de una matriz de enteros de  $n$  filas y 5 columnas. El procedimiento recibirá como parámetros: una matriz de enteros de 5 columnas, el número de filas de la matriz, y el número de columnas de la matriz.

Nota: Para elevar un valor al cuadrado, se puede utilizar la función `pow(valor,2)` de la librería `math.h`

```
for (i=0 ; i<dm ; i++)
```

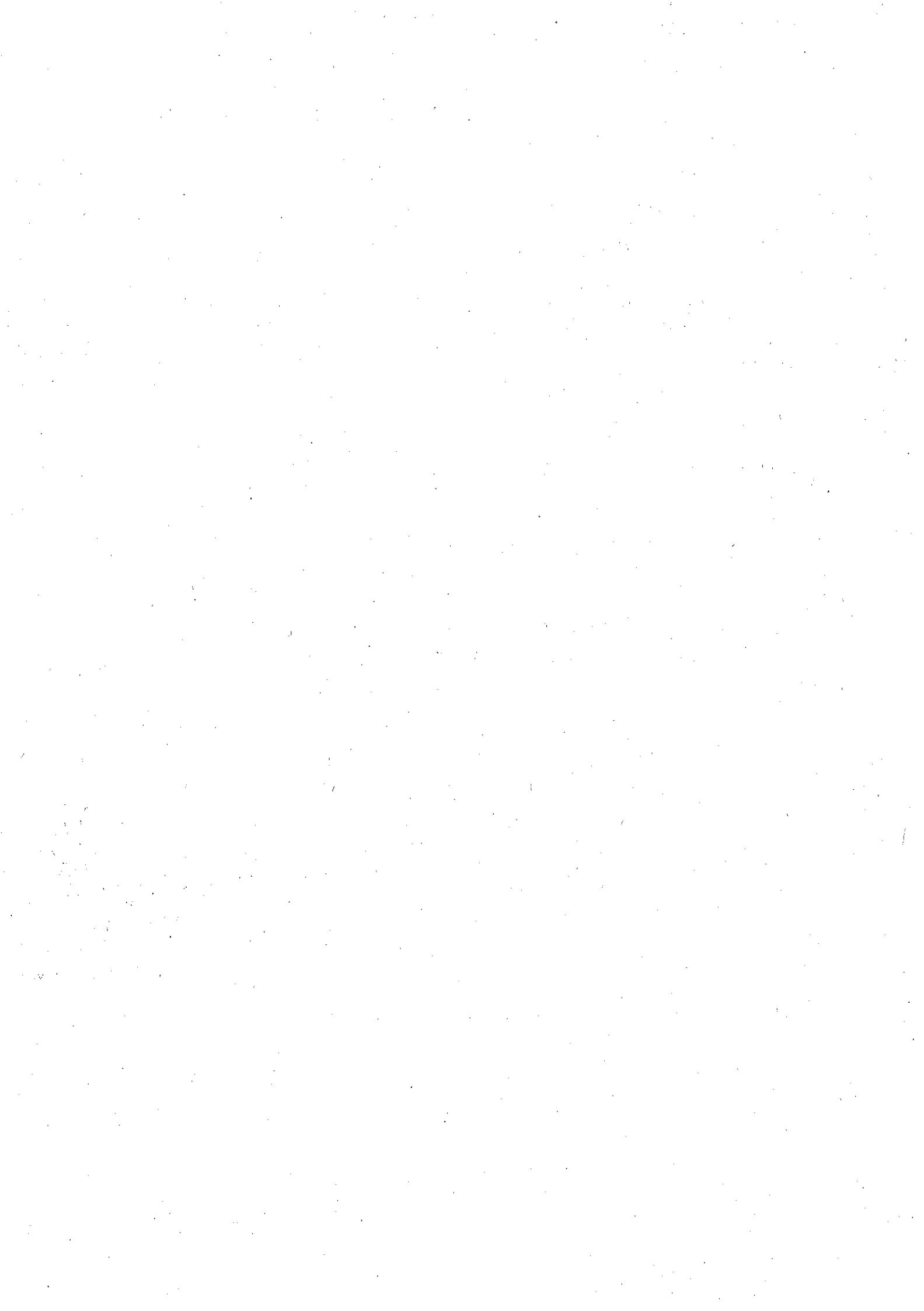
```
 while (i != 0)
 {
 i++
 }
```

```
 cont = tree;
```

```
int main()
```

```
{
 Tfront();
```

```
 cout << "Introduza seu nome
 cin
 cout
 cin
 cout
```



**Fundamentos de Informática (103806) Curso 2019-2020**  
**Examen Parcial (11 de Noviembre de 2019)**

**Nombre estudiante:** Amy Nogueras Dunn

**NIU:** A259273

Importante: Recuerde que hay que dar las mejores soluciones posibles en cada ejercicio. Además de funcionar correctamente, el código debe estar bien programado (código claro, con las instrucciones más adecuadas, sin operaciones ni variables innecesarias, etc.)

[Las funciones y procedimientos de las preguntas de la 1 a la 6 forman parte de un único programa y, por tanto, debe haber coherencia entre sus definiciones, y la forma en que se utilizan en otras preguntas. El contexto del programa a desarrollar, lo tenéis a continuación:

AgroFruits, una cooperativa agrícola, quiere almacenar datos y hacer estadísticas sobre su producción de manzanas para mejorar la calidad de su producción. La producción de manzanas se distribuye a partir de lotes que vienen identificados por un número (entero), y para cada lote se quiere guardar el diámetro medio de las manzanas del lote y su nivel de calidad según un código numérico (1,2,3,4).

Esta información se guardará en tres tablas (arrays) (una para los identificadores, una para las medidas de los diámetros y una para los niveles de calidad). La información de un mismo lote se guarda en la misma posición de las tres tablas, es decir, el identificador de un lote, su diámetro medio y su nivel de calidad se guardan en el mismo índice de las tablas correspondientes.

En total, la cooperativa quiere guardar los datos de 100 lotes de manzanas.

**Ejercicio 1 (1 punto)**

Hacer una función llamada CalidadLote para clasificar un lote en uno de los niveles de calidad marcados (1, 2, 3 ó 4). La función recibirá el valor del diámetro medio del lote (en cms) y devolverá el nivel numérico de calidad del lote según el siguiente criterio:

- nivel 1: si el diámetro medio es inferior a 5 cms.
- nivel 2: si el diámetro medio es superior o igual a 5 cms e inferior a 7 cms.
- nivel 3: si el diámetro medio es superior o igual a 7 cm e inferior a 9 cms.
- nivel 4: si el diámetro medio es superior o igual a 9 cms.

NOTA: Considere que los valores de entrada son positivos (no hay que hacer la comprobación).

$$815 - 300 = \underline{\underline{515}}$$

$$2 + 71 = 73$$

$$\text{Resul:ent} = 73$$

**Ejercicio 2 (1 punto)**

Hacer una función llamada EntrarDatosLote que pedirá los datos de un lote de manzanas y los guardará en las tablas correspondientes. La función recibirá como parámetros:

- las tres tablas con los identificadores, los diámetros medios y los niveles de calidad
- un entero con la dimensión de las tablas anteriores (un solo valor)
- la posición de la tabla donde se quieren guardar los datos que se introduzcan

La función comprobará que la posición en la que desea guardar los datos sea un índice correcto (no inferior a cero, ni superior al último índice válido de las tablas). Si la posición es incorrecta, la función devolverá -1 y no se hará nada más.

Si la posición es correcta, se pedirá al usuario que introduzca el valor del identificador del lote y el diámetro medio de las manzanas del lote, y se guardarán en la posición indicada de las tablas correspondientes.

En ambos casos, si se introduce un valor que no es positivo, se mostrará el mensaje "Error: Valor incorrecto" y se volverá a pedir el valor. Este proceso se repetirá hasta que se introduzcan los valores correctamente.

Una vez introducidas correctamente los datos anteriores, se calculará el nivel de calidad del lote (utilice la función calidadLote del ej.1) y se guardará en la tabla correspondiente.

Si se han introducido todos los datos correctamente, la función devolverá 0.

**Ejercicio 3 (1 punto)**

Hacer un procedimiento llamado `DistribucionPorNiveles` que calcule cuántos lotes hay de cada nivel de calidad y lo guarde en una tabla (array). El procedimiento recibirá como parámetros: una tabla con los niveles de calidad de los lotes, la dimensión de la tabla anterior (la pasada como primer parámetro), y una tabla donde se guardará el número de lotes de cada nivel de calidad en una posición diferente.

NOTA: La tabla donde se guardara el número de lotes de cada nivel (tercer parámetro) estará inicializada a 0 (no hay que inicializarla, en este procedimiento).

**Ejercicio 5 (1 punto)**

Hacer una función llamada `Estadistica` que calcule y devuelva algunos estadísticos (media y varianza) calculados sobre una tabla de valores reales. La función recibirá como parámetros:

- una tabla de valores reales
- un entero con la dimensión de la tabla anterior (la tabla pasada como primer parámetro)
- dos variables reales para guardar la media y la varianza de los valores de la tabla

El valor de retorno de la función será -1 si la dimensión entrada como segundo parámetro es incorrecta (más pequeña que 1) y por tanto no se pueden calcular los estadísticos. Si la dimensión es correcta y se pueden calcular correctamente la media y la varianza, la función devolverá 0.

Nota: Para calcular la varianza de un conjunto de datos, debe seguir la fórmula:

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

donde  $x_i$  son los elementos del conjunto de datos,  $\mu$  es la media de los datos y  $N$  es el número de elementos del conjunto de datos.

Para elevar un valor al cuadrado, se puede utilizar la función `pow (valor, 2)` de la librería `math.h`.

**Ejercicio 4 (1 punto)**

Hacer un procedimiento llamado `ConsultaPorNivel` que permita visualizar los datos de todos los lotes de un determinado nivel de calidad. El procedimiento recibirá como parámetros:

- las tres tablas con los identificadores, los diámetros medios y los niveles de calidad
- un entero con la dimensión de las tablas anteriores (un solo valor)

El procedimiento pedirá al usuario que introduzca el nivel de calidad a consultar. A continuación, el procedimiento mostrará por pantalla los datos (identificador y diámetro medio) de los lotes que tengan el nivel de calidad indicado por el usuario. Los datos de un mismo lote se mostrarán en una misma línea (p.ej. "ID: 1234 diámetro: 5.6cms").

Si se introduce un nivel de calidad que no es válido (niveles válidos: 1, 2, 3, 4), se mostrará el mensaje "Error: Nivel de calidad inexistente" y se saldrá del procedimiento.

**Ejercicio 6 (2.5 puntos)**

Hacer un programa completo (declaraciones globales y función main() ) que siga los siguientes pasos:

1. Declarar (donde corresponda) las constantes necesarias para que el programa sea fácilmente modificable.
2. Declarar (donde corresponda) tres estructuras de datos denominadas identificadores, medidas y niveles para guardar los identificadores de los lotes de manzanas a estudiar, los valores de sus diámetros medios y sus niveles de calidad, respectivamente.
3. Declarar (donde corresponda) una estructura de datos llamada distribucion para guardar el número de lotes que hay de cada nivel de calidad.
4. Suponed que tienes implementado un procedimiento definido como:

```
void InicializarTabla (int tabla, int nElementos);
```

que permite inicializar a cero una tabla de enteros de tamaño nElementos. Utilizar (NO IMPLEMENTAR) este procedimiento para inicializar a cero la tabla distribucion.
5. Utilizar el procedimiento EntrarDatosLote del ej. 2, para pedir los datos de TODOS los lotes y guardarlas en las estructuras de datos identificadores, medidas y niveles.
6. Mostrar un mensaje indicando que tiene 4 opciones disponibles (1-Distribución por calidades, 2- Consulta de lotes por grupo, 3-Estadísticos, 4-Salir) y pedir al usuario que introduzca una de las opciones.
7. Leer la opción elegida.
8. Si la opción elegida es 1:
  - 8.1. Llamar al procedimiento DistribucionPorNiveles del ej. 3 para calcular cuántos lotes hay de cada nivel de calidad.
  - 8.2. Mostrar por pantalla, cuántos lotes hay de cada nivel de calidad con mensajes del estilo: "Nivel 1: 12 lotes"
9. Si la opción elegida es 2, llamar al procedimiento ConsultaPorNivel del ej. 4 para visualizar por pantalla los datos de todos los lotes de un nivel de calidad concreto.
10. Si la opción elegida es 3:
  - 10.1. Llamar al procedimiento Estadistica del ej. 5 para calcular la media y la varianza de las medidas (diámetros) de los lotes de manzanas.
  - 10.2. Si la función devuelve un error, mostrar el mensaje "Error: No se han podido calcular los estadísticos". En caso contrario, mostrar por pantalla los valores de la media y la varianza.
11. Si la opción elegida no es ninguna de las 4 disponibles, mostrar el mensaje "Opción incorrecta".
12. Repetir los pasos del 6 al 11, hasta que el usuario seleccione la opción de salir.

**Ejercicio 7 (0.5 puntos)**

Declarar un nuevo tipo de datos, TIote, como un registro con los campos: Ident\_Lote (valor entero), Diametro (valor real), Calidad (valor entero) y Origen\_Lote (cadena de 50 caracteres). Hacer un procedimiento llamado llenarIote que reciba como parámetro un registro del tipo TIote y llene sus campos a partir de la información entrada por teclado.

**Ejercicio 9 (1 punto)**

Hacer una función llamada BuscarPrimerCero que reciba como parámetros un vector de enteros y un valor entero con el número de elementos del vector. La función debe calcular y devolver en qué índice del vector está el primer elemento igual a cero. Si el vector no contiene ningún cero, devolverá -1.

**Ejercicio 8 (1 punto)**

Hacer un procedimiento llamado CuadradoMatriz que eleve al cuadrado todos los elementos de una matriz de enteros de n filas y 5 columnas. El procedimiento recibirá como parámetros: una matriz de enteros de 5 columnas, el número de filas de la matriz, y el número de columnas de la matriz.

Nota: Para elevar un valor al cuadrado, se puede utilizar la función pow(valor,2) de la librería math.h

# Fundamentos de Informática (103806) Curso 2019-2020

Examen Parcial (11 de Noviembre de 2019)

Nombre estudiante: Dirk K. Whelton Metz

NIU: Q663407

Importante: Recuerde que hay que dar las mejores soluciones posibles en cada ejercicio. Además de funcionar correctamente, el código debe estar bien programado (código claro, con las instrucciones más adecuadas, sin operaciones ni variables innecesarias, etc.)

Las funciones y procedimientos de las preguntas de la 1 a la 6 forman parte de un único programa y, por tanto, debe haber coherencia entre sus definiciones, y la forma en que se utilizan en otras preguntas. El contexto del programa a desarrollar, lo tenéis a continuación:

AgroFruits, una cooperativa agrícola, quiere almacenar datos y hacer estadísticas sobre su producción de manzanas para mejorar la calidad de su producción. La producción de manzanas se distribuye a partir de lotes que vienen identificados por un número (entero), y para cada lote se quiere guardar el diámetro medio de las manzanas del lote y su nivel de calidad según un código numérico (1,2,3,4).

Esta información se guardará en tres tablas (arrays) (una para los identificadores, una para las medidas de los diámetros y una para los niveles de calidad). La información de un mismo lote se guarda en la misma posición de las tres tablas, es decir, el identificador de un lote, su diámetro medio y su nivel de calidad se guardan en el mismo índice de las tablas correspondientes.

En total, la cooperativa quiere guardar los datos de 100 lotes de manzanas.

## Ejercicio 1 (1 punto)

Hacer una función llamada CalidadLote para clasificar un lote en uno de los niveles de calidad marcados (1, 2, 3 ó 4). La función recibirá el valor del diámetro medio del lote (en cms) y devolverá el nivel numérico de calidad del lote según el siguiente criterio:

- nivel 1: si el diámetro medio es inferior a 5 cms.
- nivel 2: si el diámetro medio es superior o igual a 5 cms e inferior a 7 cms.
- nivel 3: si el diámetro medio es superior o igual a 7 cm e inferior a 9 cms.
- nivel 4: si el diámetro medio es superior o igual a 9 cms.

NOTA: Considere que los valores de entrada son positivos (no hay que hacer la comprobación).

$$10 + 5 = 15$$
$$15 - 2 = \underline{13}$$

## Ejercicio 2 (1 punto)

Hacer una función llamada EntrarDatosLote que pedirá los datos de un lote de manzanas y los guardará en las tablas correspondientes. La función recibirá como parámetros:

- las tres tablas con los identificadores, los diámetros medios y los niveles de calidad
- un entero con la dimensión de las tablas anteriores (un solo valor)

La función comprobará que la posición en la que desea guardar los datos sea un índice correcto (no inferior a cero, ni superior al último índice válido de las tablas). Si la posición es incorrecta, la función devolverá -1 y no se hará nada más.

Si la posición es correcta, se pedirá al usuario que introduzca el valor del identificador del lote y el diámetro medio de las manzanas del lote, y se guardarán en la posición indicada de las tablas correspondientes.

En ambos casos, si se introduce un valor que no es positivo, se mostrará el mensaje "Error: Valor incorrecto" y se volverá a pedir el valor. Este proceso se repetirá hasta que se introduzcan los valores correctamente.

Una vez introducidas correctamente los datos anteriores, se calculará el nivel de calidad del lote (utilice la función CalidadLote del ej.1) y se guardará en la tabla correspondiente.

Si se han introducido todos los datos correctamente, la función devolverá 0.

$$200 + 78 = 278$$

$$278 - 23 = 255$$

Resultado : 255

**Ejercicio 3 (1 punto)**

Hacer un procedimiento llamado `DistribucionPorNiveles` que calcule cuántos lotes hay de cada nivel de calidad y lo guarde en una tabla (array). El procedimiento recibirá como parámetros: una tabla con los niveles de calidad de los lotes, la dimensión de la tabla anterior (la pasada como primer parámetro), y una tabla donde se guardará el número de lotes de cada nivel de calidad en una posición diferente.

NOTA: La tabla donde se guardará el número de lotes de cada nivel (tercer parámetro) estará inicializada a 0 (no hay que inicializarla, en este procedimiento).

**Ejercicio 5 (1 punto)**

Hacer una función llamada `Estadistica` que calcule y devuelva algunos estadísticos (media y varianza) calculados sobre una tabla de valores reales. La función recibirá como parámetros:

- una tabla de valores reales
- un entero con la dimensión de la tabla anterior (la tabla pasada como primer parámetro)
- dos variables reales para guardar la media y la varianza de los valores de la tabla

El valor de retorno de la función será -1 si la dimensión entrada como segundo parámetro es incorrecta (más pequeña que 1) y por tanto no se pueden calcular los estadísticos. Si la dimensión es correcta y se pueden calcular correctamente la media y la varianza, la función devolverá 0.

Note: Para calcular la varianza de un conjunto de datos, debe seguir la fórmula:

$$\text{varianza} = \frac{1}{N-1} \sum_i (x_i - \mu)^2$$
 donde  $x_i$  son los elementos del conjunto de datos,  $\mu$  es la media de los datos y  $N$  es el número de elementos del conjunto de datos.

Para elevar un valor al cuadrado, se puede utilizar la función `pow (valor, 2)` de la librería `math.h`.

**Ejercicio 4 (1 punto)**

Hacer un procedimiento llamado `ConsultaPorNivel` que permita visualizar los datos de todos los lotes de un determinado nivel de calidad. El procedimiento recibirá como parámetros:

- las tres tablas con los identificadores, los diámetros medios y los niveles de calidad
- un entero con la dimensión de las tablas anteriores (un solo valor)

El procedimiento pedirá al usuario que introduzca el nivel de calidad a consultar. A continuación, el procedimiento mostrará por pantalla los datos (identificador y diámetro medio) de los lotes que tengan el nivel de calidad indicado por el usuario. Los datos de un mismo lote se mostrarán en una misma línea (p. ej. "ID: 1234 diámetro: 5.6cm").

Si se introduce un nivel de calidad que no es válido (niveles válidos: 1, 2, 3, 4), se mostrará el mensaje "Error: Nivel de calidad inexistente" y se saldrá del procedimiento.

**Ejercicio 6 (2.5 puntos)**

Hacer un programa completo (declaraciones globales y función main() ) que siga los siguientes pasos:

1. Declarar (donde corresponda) las constantes necesarias para que el programa sea fácilmente modificable.
2. Declarar (donde corresponda) tres estructuras de datos denominadas `identificadores`, `medidas` y `niveles` para guardar los identificadores de los lotes de manzanas a estudiar, los valores de sus diámetros medios y sus niveles de calidad, respectivamente.
3. Declarar (donde corresponda) una estructura de datos llamada `distribucion` para guardar el número de lotes que hay de cada nivel de calidad.
4. Suponed que tenéis implementado un procedimiento definido como:  
`void InicializarTabla (int tabla, int nElementos);`  
que permite inicializar a cero una tabla de enteros de tamaño `nElementos`. Utilizar (NO IMPLEMENTAR) este procedimiento para inicializar a cero la tabla `distribucion`.
5. Utilizar el procedimiento `EntrarDatosLote` del ej. 2, para pedir los datos de `TODOS` los lotes y guardarlas en las estructuras de datos `identificadores`, `medidas` y `niveles`.
6. Mostrar un mensaje indicando que tiene 4 opciones disponibles (1-Distribución por calidad, 2- Consulta de lotes por grupo, 3-Estadísticos, 4-Salir) y pedir al usuario que introduzca una de las opciones.
7. Leer la opción elegida.
8. Si la opción elegida es 1:
  - 8.1. Llamar al procedimiento `DistribucionPorNiveles` del ej. 3 para calcular cuántos lotes hay de cada nivel de calidad.
  - 8.2. Mostrar por pantalla, cuántos lotes hay de cada nivel de calidad con mensajes del estilo:  
"Nivel 1: 12 lotes"
9. Si la opción elegida es 2, llamar al procedimiento `ConsultaPorNivel` del ej. 4 para visualizar por pantalla los datos de todos los lotes de un nivel de calidad concreto.
10. Si la opción elegida es 3:
  - 10.1. Llamar al procedimiento `Estadistica` del ej. 5 para calcular la media y la varianza de las medidas (diámetros) de los lotes de manzanas.
  - 10.2. Si la función devuelve un error, mostrar el mensaje "Error: No se han podido calcular los estadísticos". En caso contrario, mostrar por pantalla los valores de la media y la varianza.
11. Si la opción elegida no es ninguna de las 4 disponibles, mostrar el mensaje "Opción incorrecta".
12. Repetir los pasos del 6 al 11, hasta que el usuario seleccione la opción de salir.

**Ejercicio 7 (0.5 puntos)**

Declarar un nuevo tipo de datos, `TLote`, como un registro con los campos: `Ident_lote` (valor entero), `Diámetro` (valor real), `Calidad` (valor entero) y `Origen_lote` (cadena de 50 caracteres). Hacer un procedimiento llamado `LlenarLote` que reciba como parámetro un registro del tipo `TLote` y llene sus campos a partir de la información entrada por teclado.

**Ejercicio 9 (1 punto)**

Hacer una función llamada `BuscarPrimerCero` que reciba como parámetros un vector de enteros y un valor entero con el número de elementos del vector. La función debe calcular y devolver en qué índice del vector está el primer elemento igual a cero. Si el vector no contiene ningún cero, devolverá -1.

**Ejercicio 8 (1 punto)**

Hacer un procedimiento llamado `CuadradoMatriz` que eleve al cuadrado todos los elementos de una matriz de enteros de  $n$  filas y 5 columnas. El procedimiento recibirá como parámetros: una matriz de enteros de 5 columnas, el número de filas de la matriz, y el número de columnas de la matriz.

Nota: Para elevar un valor al cuadrado, se puede utilizar la función `pow(valor,2)` de la librería `math.h`

**Alumnos ue informática I usos) uniu avizcau  
Examen Parcial (11 de Noviembre de 2019) Grupo:  
Nombre estudiante: Añón Álvarez Quijano NIU: 5117969**

Importante: Recuerde que hay que dar las mejores soluciones posibles en cada ejercicio. Además de funcionar correctamente, el código debe estar bien programado (código claro, con las instrucciones más adecuadas, sin operaciones ni variables innecesarias, etc.)

Las funciones y procedimientos de las preguntas de la 1 a la 6 forman parte de un único programa y, por tanto, debe haber coherencia entre sus definiciones, y la forma en que se utilizan en otras preguntas. El contexto del programa a desarrollar, lo tenéis a continuación:

BeautyFlors, una empresa dedicada al cultivo de flores, quiere almacenar datos y hacer estadísticas sobre su producción de rosas para mejorar la calidad de su producción. La producción de rosas se distribuye a partir de lotes que vienen identificados por un número (entero), y para cada lote se quiere guardar el número medio de pétalos del lote y un código de venta numérico (10,11,12,13) que indica el precio de venta del lote.

Esta información se guardará en tres tablas (arrays) (una para los identificadores, una para las medidas del número de pétalos y una para el código de venta). La información de un mismo lote se guarda en la misma posición de las tres tablas, es decir, el identificador de un lote, su número medio de pétalos y su código de venta se guardan en el mismo índice de las tablas correspondientes. En total, la empresa quiere guardar los datos de 150 lotes de flores.

**Ejercicio 1 (1 punto)**

Hacer una función llamada CódigoVentaLote para clasificar un lote en uno de los códigos de venta marcados (10, 11, 12 ó 13). La función recibirá el valor del número medio de pétalos del lote y devolverá el código numérico de venta del lote según el siguiente criterio:

- código 10: si el número medio de pétalos es inferior a 15
- código 11: si el número medio de pétalos es superior o igual a 15 cms e inferior a 25
- código 12: si el número medio de pétalos es superior o igual a 25 cm e inferior a 35
- código 13: si el número medio de pétalos es superior o igual a 35

NOTA: Considere que los valores de entrada son positivos (no hay que hacer la comprobación).

$$30 \times 2 = \underline{60}$$
$$60 + 100 = \underline{160}$$

Hacer una función llamada EntrarDatosLote que pedirá los datos de un lote de rosas y los guardará en las tablas correspondientes. La función recibirá como parámetros:

- las tres tablas con los identificadores, los números medios de pétalos y los códigos de precio
- un entero con la dimensión de las tablas anteriores (un solo valor)
- la posición de la tabla donde se quieren guardar los datos que se introduzcan

La función comprobará que la posición en la que desea guardar los datos sea un índice correcto (no inferior a cero, ni superior al último índice válido de las tablas). Si la posición es incorrecta, la función devolverá -1 y no se hará nada más.

Si la posición es correcta, se pedirá al usuario que introduzca el valor del identificador del lote y el número medio de pétalos del lote de rosas, y se guardarán en la posición indicada de las tablas correspondientes.

En ambos casos, si se introduce un valor que no es positivo, se mostrará el mensaje "Error: Valor Incorrecto" y se volverá a pedir el valor. Este proceso se repetirá hasta que se introduzcan los valores correctamente.

Una vez introducidas correctamente los datos anteriores, se calculará el código de venta del lote (utilice la función CódigoVentaLote del ej. 1) y se guardará en la tabla correspondiente.

Si se han introducido todos los datos correctamente, la función devolverá 0.

$$700 - 200 = 500$$

$$500 + 83 = 583$$

$$\text{Resultado} = 583$$

Ejercicio 3 (1 punto)

Hacer un procedimiento llamado DistribucionPorCódigos que calcule cuantos lotes hay de cada código de venta y lo guarde en una tabla (array). El procedimiento recibirá como parámetros: una tabla con los códigos de venta de los lotes, la dimensión de la tabla anterior (la pasada como primer parámetro), y una tabla donde se guardará el número de lotes de cada código de venta en una posición diferente.

NOTA: La tabla donde se guardará el número de lotes de cada código de venta (tercer parámetro) estará inicializada da a 0 (no hay que inicializarla, en este procedimiento).

Ejercicio 5 (1 punto)

Hacer una función llamada Estadística que calcule y devuelva algunos estadísticos (máximo y mínimo) calculados sobre una tabla de valores reales. La función recibirá como parámetros:

- una tabla de valores reales
- un entero con la dimensión de la tabla anterior (la tabla pasada como primer parámetro)
- dos variables reales para guardar el máximo y el mínimo de los valores de la tabla

El valor de retorno de la función será -1 si la dimensión entrada como segundo parámetro es incorrecta (más pequeña que 1) y por tanto no se pueden calcular los estadísticos. Si la dimensión es correcta y se pueden calcular correctamente el máximo y el mínimo, la función devolverá 0.

#### Ejercicio 4 (1 punto)

Hacer un procedimiento llamado ConsultaPorCódigo que permita visualizar los datos de todos los lotes con un determinado código de venta. El procedimiento recibirá como parámetros:

- las tres tablas con los identificadores, los números medios péntilos y los códigos de venta
- un entero con la dimensión de las tablas anteriores (un solo valor)

El procedimiento pedirá al usuario que introduzca el código de venta a consultar. A continuación, el procedimiento mostrará por pantalla los datos (identificador y numero medio de péntilos) de los lotes que tengan el código de venta indicado por el usuario. Los datos de un mismo lote se mostrarán en una misma línea (p.ej. "ID:1234 num medio de pentilos: 32. 6").

Si se introduce un código de venta que no es válido (códigos válidos: 10, 11, 12, 13), se mostrará el mensaje "Error: Código de venta inexistente" y se saldrá del procedimiento.

acer un programa completo (declaraciones globales y función main() ) que siga los siguientes pasos:

1. Declarar (donde corresponda) las constantes necesarias para que el programa sea fácilmente modificable.
2. Declarar (donde corresponda) tres estructuras de datos denominadas identificadores, medidas y códigos para guardar los identificadores de los lotes de rosas a estudiar, los valores de sus números medios de pétalos y sus códigos de venta, respectivamente.
3. Declarar (donde corresponda) una estructura de datos llamada distribucion para guardar el número de lotes que hay de cada código de venta.
4. Suponed que tenéis implementado un procedimiento definido como:  

```
void InicializarTabla (int tabla[], int nElementos);
```

que permite inicializar a cero una tabla de enteros de tamaño nElementos. Utilice (NO implementado) este procedimiento para inicializar a cero la tabla distribucion.
5. Utilizar la función EntrarDatosLote del ej. 2, para pedir los datos de TODOS los lotes y guardarlos en las estructuras de datos identificadores, medidas y códigos.
6. Mostrar un mensaje indicando que tiene 4 opciones disponibles (1-Distribución por códigos de venta, 2-Consulta de lotes por código de venta, 3-Estadísticos, 4-Salir) y pedir al usuario que introduzca una de las opciones.
7. Leer la opción elegida.
8. Si la opción elegida es 1:
- 8.1. Llamar al procedimiento DistribucionPorCódigos del ej. 3 para calcular cuántos lotes hay de cada código de venta.
- 8.2. Mostrar por pantalla, cuántos lotes hay de cada código de venta con mensajes del estilo:  
"Código 10: 12 lotes"
9. Si la opción elegida es 2, llamar al procedimiento ConsultaPorCodigo del ej. 4 para visualizar por pantalla los datos de todos los lotes de un código de venta concreto.
10. Si la opción elegida es 3:
- 10.1. Llamar al procedimiento Estadistica del ej. 5 para calcular el máximo y el mínimo de las medidas (número medio de pétalos) de los lotes de rosas.
- 10.2. Si la función devuelve un error, mostrar el mensaje "Error: No se han podido calcular los estadísticos". En caso contrario, mostrar por pantalla los valores del máximo y el mínimo.
11. Si la opción elegida no es ninguna de las 4 disponibles, mostrar el mensaje "Opción incorrecta".
12. Repetir los pasos del 6 al 11, hasta que el usuario seleccione la opción de salir.

**Ejercicio 7 (0,5 puntos)**  
Declarar un nuevo tipo de datos, `TLote`, como un registro con los campos: `Ident_Lote` (valor entero), `Num_medio_petalos` (valor real), `Codigo_venta` (valor entero) y `Origen_lote` (cadena de 50 caracteres).

Hacer un procedimiento llamado `InicializarLote` que reciba como parámetro un registro del tipo `TLote` e inicialice sus campos con valor cero en los campos numéricos, y valor `" "` en el campo de tipo cadena de caracteres.

**Ejercicio 9 (1 punto)**

Hacer una función llamada `BuscarPrimerImpar` que reciba como parámetros un vector de enteros y un valor entero con el número de elementos del vector. La función debe calcular y devolver en qué índice del vector está el primer elemento impar. Si el vector no contiene ningún valor impar, devolverá -1.

**Ejercicio 8 (1 punto)**

Hacer un procedimiento llamado `Doblementriz` que doble el valor de todos los elementos de una matriz de enteros de  $n$  filas y 5 columnas. El procedimiento recibirá como parámetros: una matriz de enteros de 5 columnas, el número de filas de la matriz, y el número de columnas de la matriz.

for (int i=0; i<DIM; i++)

{

~~Results[i] =~~

Results[i] = Vots(i) + f;

3 Parlamentarios

- |   |    |
|---|----|
| 1 | 43 |
| 2 | 27 |
| 3 | 50 |
| 4 | 90 |
| 5 | 21 |

if ((0.25 \* salario) > cantidad\_poblacion)

if ((0.50 \* salario) > cantidad\_poblacion)

BB + 5

~~DOE APPROVAL~~

**Alumnos de informática (IUSOUD) curso 2019-2020**  
**Examen Parcial (11 de Noviembre de 2019)**

**Nombre estudiante:** Gonzalo Rivas Herran **NIU:** 676945

Importante: Recuerde que hay que dar las mejores soluciones posibles en cada ejercicio. Además de funcionar correctamente, el código debe estar bien programado (código claro, con las instrucciones más adecuadas, sin operaciones ni variables innecesarias, etc.)

Las funciones y procedimientos de las preguntas de la 1 a la 6 forman parte de un único programa y, por tanto, debe haber coherencia entre sus definiciones, y la forma en que se utilizan en otras preguntas. El contexto del programa a desarrollar, lo tenéis a continuación:

BeautyFlors, una empresa dedicada al cultivo de flores, quiere almacenar datos y hacer estadísticas sobre su producción de rosas para mejorar la calidad de su producción. La producción de rosas se distribuye a partir de lotes que vienen identificados por un número (entero), y para cada lote se quiere guardar el número medio de pétalos del lote y un código de venta numérico (10,11,12,13) que indica el precio de venta del lote.

Esta información se guardará en tres tablas (arrays) (una para los identificadores, una para las medidas del número de pétalos y una para el código de venta). La información de un mismo lote se guarda en la misma posición de las tres tablas, es decir, el identificador de un lote, su número medio de pétalos y su código de venta se guardan en el mismo índice de las tablas correspondientes.

En total, la empresa quiere guardar los datos de 150 lotes de flores.

**Ejercicio 1 (1 punto)**

Hacer una función llamada CódigoVentaLote para clasificar un lote en uno de los códigos de venta marcados (10, 11, 12 ó 13). La función recibirá el valor del número medio de pétalos del lote y devolverá el código numérico de venta del lote según el siguiente criterio:

- código 10: si el número medio de pétalos es inferior a 15
- código 11: si el número medio de pétalos es superior o igual a 15 cms e inferior a 25
- código 12: si el número medio de pétalos es superior o igual a 25 cm e inferior a 35
- código 13: si el número medio de pétalos es superior o igual a 35

**NOTA:** Considere que los valores de entrada son positivos (no hay que hacer la comprobación).

$$442 - 361 = 381$$
$$381 \div 3 = \boxed{127}$$

$$120 + 58 = 178$$

$$178 \times 5 = 890$$

$$\text{Renta} : 890$$

Hacer una función llamada EntrarDatosLote que pedirá los datos de un lote de rosas y los guardará en las tablas correspondientes. La función recibirá como parámetros:

- las tres tablas con los identificadores, los números medios de pétalos y los códigos de precio
- la posición de la tabla donde se quieren guardar los datos que se introduzcan

La función comprobará que la posición en la que desea guardar los datos sea un índice correcto (no inferior a cero, ni superior al último índice válido de las tablas). Si la posición es incorrecta, la función devolverá -1 y no se hará nada más.

Si la posición es correcta, se pedirá al usuario que introduzca el valor del identificador del lote y el número medio de pétalos del lote de rosas, y se guardarán en la posición indicada de las tablas correspondientes.

En ambos casos, si se introduce un valor que no es positivo, se mostrará el mensaje "Error: Valor incorrecto" y se volverá a pedir el valor. Este proceso se repetirá hasta que se introduzcan los valores correctamente.

Una vez introducidas correctamente los datos anteriores, se calculará el código de venta del lote (utilice la función CódigoVentaLote del ej.1) y se guardará en la tabla correspondiente.

Si se han introducido todos los datos correctamente, la función devolverá 0.

Ejercicio 4 (1 punto)

Hacer un procedimiento llamado DistribucionPorCódigos que calcule cuántos lotes hay de cada código de venta y lo guarde en una tabla (array). El procedimiento recibirá como parámetros: una tabla con los códigos de venta de los lotes, la dimensión de la tabla anterior (la pasada como primer parámetro), y una tabla donde se guardará el número de lotes de cada código de venta en una posición diferente.

NOTA: La tabla donde se guardará el número de lotes de cada código de venta (tercer parámetro) estará inicializada da a 0 (no hay que inicializarla, en este procedimiento).

Ejercicio 5 (1 punto)

Hacer una función llamada Estadística que calcule y devuelva algunos estadísticos (máximo y mínimo) calculados sobre una tabla de valores reales. La función recibirá como parámetros:

- una tabla de valores reales
- un entero con la dimensión de la tabla anterior (la tabla pasada como primer parámetro)
- dos variables reales para guardar el máximo y el mínimo de los valores de la tabla

El valor de retorno de la función será -1 si la dimensión entrada como segundo parámetro es incorrecta (más pequeña que 1) y por tanto no se pueden calcular los estadísticos. Si la dimensión es correcta y se pueden calcular correctamente el máximo y el mínimo, la función devolverá 0.

Ejercicio 4 (1 punto)

Hacer un procedimiento llamado ConsultaPor Código que permita visualizar los datos de todos los lotes con un determinado código de venta. El procedimiento recibirá como parámetros:

- las tres tablas con los identificadores, los números medios pétalos y los códigos de venta
- un entero con la dimensión de las tablas anteriores (un solo valor)

El procedimiento pedirá al usuario que introduzca el código de venta a consultar. A continuación, el procedimiento mostrará por pantalla los datos (identificador y número medio de pétalos) de los lotes que tengan el código de venta indicado por el usuario. Los datos de un mismo lote se mostrarán en una misma linea (p.ej. "ID:1234 num medio de petalos: 32.6").

Si se introduce un código de venta que no es válido (códigos válidos: 10, 11, 12, 13), se mostrará el mensaje "Error:Código de venta inexistente" y se saldrá del procedimiento.

cer un programa completo (declaraciones globales y función main() ) que siga los siguientes pasos:

1. Declarar (donde corresponda) las constantes necesarias para que el programa sea fácilmente modificable.
2. Declarar (donde corresponda) tres estructuras de datos denominadas identificadores, medidas y códigos para guardar los identificadores de los lotes de rosas a estudiar, los valores de sus números medios de pétalos y sus códigos de venta, respectivamente.
3. Declarar (donde corresponda) una estructura de datos llamada distribucion para guardar el número de lotes que hay de cada código de venta.
4. Suponed que tenéis implementado un procedimiento definido como:

```
void InicializarTabla (int tabla[], int nElementos);
```

que permite inicializar a cero una tabla de enteros de tamaño nElementos. Utilice (NO implementado) este procedimiento para inicializar a cero la tabla distribucion.
5. Utilizar la función EntrarDatosLote del ej. 2, para pedir los datos de TODOS los lotes y guardarlo en las estructuras de datos identificadores, medidas y códigos.
6. Mostrar un mensaje indicando que tiene 4 opciones disponibles (1-Distribución por códigos de venta, 2-Consulta de lotes por código de venta, 3-Estadísticos, 4-Salir) y pedir al usuario que introduzca una de las opciones.
7. Leer la opción elegida.
8. Si la opción elegida es 1:
  - 8.1. Llamar al procedimiento DistribucionPorCódigos del ej. 3 para calcular cuántos lotes hay de cada código de venta.
  - 8.2. Mostrar por pantalla, cuántos lotes hay de cada código de venta con mensajes del estilo: "Código 10: 12 Lotes"
9. Si la opción elegida es 2, llamar al procedimiento ConsultaPorCodigo del ej. 4 para visualizar por pantalla los datos de todos los lotes de un código de venta concreto.
10. Si la opción elegida es 3:
  - 10.1. Llamar al procedimiento Estadistica del ej. 5 para calcular el máximo y el mínimo de las medidas (número medio de pétalos) de los lotes de rosas.
  - 10.2. Si la función devuelve un error, mostrar el mensaje "Error: No se han podido calcular los estadísticos". En caso contrario, mostrar por pantalla los valores del máximo y el mínimo.
11. Si la opción elegida no es ninguna de las 4 disponibles, mostrar el mensaje "Opción incorrecta".
12. Repetir los pasos del 6 al 11, hasta que el usuario seleccione la opción de salir.

**Ejercicio 7 (0.5 puntos)**

Declarar un nuevo tipo de datos, `Tlote`, como un registro con los campos: `Ident_Lote` (valor entero), `Num_medio_petalos` (valor real), `Codigo_venta` (valor entero) y `Origen_lote` (cadena de 50 caracteres).

Hacer un procedimiento llamado `InicializarLote` que reciba como parámetro un registro del tipo `Tlote` e inicialice sus campos con valor cero en los campos numéricos, y valor "" en el campo de tipo cadena de caracteres.

**Ejercicio 9 (1 punto)**

Hacer una función llamada `BuscarPrimerImpar` que reciba como parámetros un vector de enteros y un valor entero con el número de elementos del vector. La función debe calcular y devolver en qué índice del vector está el primer elemento impar. Si el vector no contiene ningún valor impar, devolverá -1.

**Ejercicio 8 (1 punto)**

Hacer un procedimiento llamado `DobleMatriz` que doble el valor de todos los elementos de una matriz de enteros de  $n$  filas y 5 columnas. El procedimiento recibirá como parámetros: una matriz de enteros de 5 columnas, el número de filas de la matriz, y el número de columnas de la matriz.

Value (int min, int max) {

1496770

int vot;  
if (vot < min || vot > max)  
cout << "Error: " << min << endl;  
else if (vot < max)  
cout << "Error: " << endl;

while (vot < min || vot > max){  
cout << "Error: " << endl;  
cin >> vot;  
}

typedef struct {

{ char name[15];  
int vot;  
int num;  
int vot;  
} TVotes;

typedef struct

{ int num;  
int vot;

} TResult;

p = 1;

1496770  
1496770  
1496770  
1496770

1496770

$\text{if } ((y \& z) \geq 0) \text{ then } ((x_1 - y_1) \leq 0)$

$$\text{aux} = \cancel{\text{pow}(x_1 - y_1)} + \text{pow}$$

卷之三

$$\frac{10000}{100} \times \frac{100}{100} = (50000)$$

100000, 100000  
public solution

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardara en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- la posició de la taula on es volen guardar les dades que s'introduixin
- un enter amb la dimensió de les taules anteriors (un sol valor)

**Important:** Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.) Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherentia entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el teniu a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter) i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una pels identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

#### **Exercici 1 (1 punt)**

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

**NOTA:** Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$2000 - 400 = 1600$$

$$1600 \times 5 = \underline{\underline{8000}}$$

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index valid de les taules). Si la posició és incorrecta, la funció retornara -1 i no es farà res més.

Si la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardararan a la posició indicada de les taules corresponents.

En els dos casos, si s'introduceix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demandar el valor. Aquest procés es repetirà fins que s'introdueixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzant la funció CodiVendaLot de l'ex.1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$15 \times 3 = 45$$

$$45 \div 5 = 9$$

Resposta: 9

#### Exercici 4 (1 punt)

Fer un procediment anomenat `DistribucioPerCodis` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada codi de venda en una posició different.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

**Exercici 4 (1 punt)**  
Fer un procediment anomenat `ConsultaPerCodí` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introduixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i número mitja de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.6").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrara el missatge "Error: Codí de venda inexistent" i es sortirà.

#### Exercici 5 (1 punt)

Fer una funció anomenada `Estatistica` que calculi i retorni alguns estadístics (màxim, mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

er un programa complet (declaracions globals i funció main() ) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.

2. Declarar (on correspongui) tres estructures de dades anomenades identificadors,

mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.

3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.

4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.

5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.

6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.

7. Llegir l'opció escollida.

8. Si l'opció escollida és 1:

8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.

8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"

9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodí de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.

10. Si l'opció escollida és 3:

10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.

10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.

11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".

12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

**Exercici 7 (0,5 punts)**

Declarar un nou tipus de dades, TIOT, com un registre amb els camps: Ident\_lot (valor enter), Num\_mitja\_petals (valor real), Codi\_venda (valor enter) i Origen\_lot (cadena de 50 caràcters).

Fer un procediment anomenat InicialitzarIOT que rebi com a paràmetre un registre del tipus TIOT i inicialitzi els seus camps amb valor zero als camps numèrics, i valor "" al camp de tipus cadena de caràcters.

**Exercici 9 (1 punt)**

Fer una funció anomenada BuscarPrimerImparell que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

**Exercici 8 (1 punt)**

Fer un procediment anomenat DoblaMatriu que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

1- #include <stdio.h>  
 #include <conio.h>  
 #include <math.h>  
 #define Num\_Vots 135  
 int Vots [Num\_Vots];  
 int Resultats [Num\_Vots];

~~#include <math.h>~~

Diputats 135 - Vots 135

# define Min\_Vot 1  
 # define Max\_Vot 135

2- int Votar (int min, int max)

{ int vots\_vot;  
 cout << "Introdueix una votació entre " << min << " i " << max << endl;  
 cin >> vots\_vot;

~~min > i  
 max = 135~~

intisible ((vot < min) || (vot > max))

{ cout << "Error: votament fora del rang" << endl;

cout << "Introdueix una votació entre " << min << " i " << max << ":" << endl;  
 cin >> vots\_vot;

}

return vots\_vot;

}

3- void InicialitzarArray (int v[], int DIM1, int valor)

{ int i;

For (i=0; i < DIM1; i++)

{  
 } v[i];  
 }

4- int NaZeroeArray (int v[], int Dim1)

{ int i;

bool trobat=false;

do i=0

while (i < Dim1 && !trobat)

{ ~~if (v[i]==0)~~ if (trobat)

return 1;

else i++;

} if (v[i]==0)

- 5- word count (if word [Alph-Vet-Z],
- 6- if Index Minimise size-Accay (if  $\vee [ ]$ , if down) { if vector Minim,
- 7- if  $\oplus$  (if  $\neg A$ , if  $\neg B$ ) { if  $\neg$  ? ! } if  $\neg$  ? ! { if  $\neg$  ? ! }
- 8- if Computation of  $\neg$  (if  $\neg$  Accy-Demand, if  $\neg$  Satisfy-true, if  $\neg$  Satisfy-false) { if  $\neg$  Accy-Demand, if  $\neg$  Satisfy-false, if  $\neg$  Satisfy-true) { if  $\neg$  Accy-Demand, if  $\neg$  Satisfy-false, if  $\neg$  Satisfy-true) {
- 9- if (Accy-Demand, if  $\neg$  Satisfy-false, if  $\neg$  Satisfy-true) { if  $\neg$  Accy-Demand, if  $\neg$  Satisfy-false, if  $\neg$  Satisfy-true) { if  $\neg$  Accy-Demand, if  $\neg$  Satisfy-false, if  $\neg$  Satisfy-true) {
- 10- if Computation of  $\neg$  (if  $\neg$  Accy-Demand, if  $\neg$  Satisfy-false, if  $\neg$  Satisfy-true) { if  $\neg$  Accy-Demand, if  $\neg$  Satisfy-false, if  $\neg$  Satisfy-true) { if  $\neg$  Accy-Demand, if  $\neg$  Satisfy-false, if  $\neg$  Satisfy-true) {

**Examen Parcial (11 de Novembre de 2019)**

**Nom estudiant:** David Solà

**Grup:**

**NIU:** 8375438

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherència entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenu a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter) i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una per als identificadors, una per les mesures del nombre de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu nombre mitjà de pètals i el seu codi de venda es guarden al mateix index de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

**Exercici 1 (1 punt)**

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$4 + 6 = \underline{10}$$
$$10 + 100 = \underline{110}$$

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardara en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardarán a la posició indicada de les taules corresponents.

En els dos casos, si s'introduïx un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demandar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzeu la funció CodiVendaLot de l'ex. 1) i es guardarà a la taula corresponent.

Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$500 \times 2 = 1000$$

$$1000 \div 5 = 200$$

$$200 \times 10 = 2000$$

**Exercici 3 (1 punt)**  
Fer un procediment anomenat `DistribucioPerCodi` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada codi de venda en una posició diferent.

**NOTA:** La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

**Exercici 5 (1 punt)**  
Fer una funció anomenada `Estadistica` que calculi i retorna alguns estadístics (màxim, mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
  - un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
  - dos variables reals per a guardar el màxim i el mínim dels valors de la taula
- El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

**Exercici 4 (1 punt)**  
Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introdueixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador i número mitja de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p. ex. "ID: 1234 num mitja de pètals: 32.6").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codi de venda inexistent" i es sortirà.

er un programa complet (declaracions globals i funció main() ) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.

2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.

3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.

4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'elements de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribució.

5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.

6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.

7. Llegir l'opció escollida.

8. Si l'opció escollida és 1:

- 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.

- 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"

9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodis de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.

10. Si l'opció escollida és 3:

- 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.

- 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.

11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".

12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

**Exercici 7 (0,5 punts)**

Declarar un nou tipus de dades, `TLot`, com un registre amb els camps: `Ident_lot` (valor enter), `Num_mitja_petals` (valor real), `Codi_venda` (valor enter) i `Origen_lot` (cadena de 50 caràcters).

Fer un procediment anomenat `IniciarLot` que rebi com a paràmetre un registre del tipus `TLot` i inicialitzi els seus camps amb valor zero als camps numèrics, i valor `"__"` al camp de tipus cadena de caràcters.

**Exercici 9 (1 punt)**

Fer una funció anomenada `BuscarPrimerImparell` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

**Exercici 8 (1 punt)**

Fer un procediment anomenat `DobleMatriu` que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

**Examen Parcial (11 de Novembre de 2019)**

**Nom estudiant:** Eduard Recio **NIU:** 6796371

**Important:** Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha d'estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherentia entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenui a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardará en tres taules (arrays) (una pels identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

**Exercici 1 (1 punt)**

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10; si el numero mitjà de pètals és inferior a 15
- codi 11; si el numero mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12; si el numero mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13; si el numero mitjà de pètals és superior o igual a 35

**NOTA:** Considerieu que els valors d'entrada són positius (no cal fer la comprovació).

$$2 - 2 = \boxed{0}$$
$$0 + 10 = \boxed{10}$$

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- la posició de la taula on es volen guardar les dades que s'introduixin
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornarà -1 i no es farà res més.

Si la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardararan a la posició indicada de les taules corresponents.

En els dos casos, si s'introduïx un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demandar el valor. Aquest procés es repetirà fins que s'introduixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzant la funció CodiVendaLot de l'ex.1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$5 \times 5 = 25$$

$$25 - 2 = 23$$

$$\text{Resultat: } 23$$

**Exercici 3 (1 punt)**

Fer un procediment anomenat `DistribucióPerCodi` que calculi quants lôts hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lôts, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lôts de cada codi de venda en una posició diferent.

NOTA: La taula on es guardarà el número de lôts de cada codi de venda (tercer paràmetre) està inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

**Exercici 4 (1 punt)**

Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lôts amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de petals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introdueixi el codi de venda que vol consultar. A continuació, el procediment mostrara per pantalla les dades (identificador i número mitja de petals) dels lôts que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.ex. "ID: 1234 num mitja de petals: 32.5").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codí de venda inexistent" i es sortirà.

**Exercici 5 (1 punt)**

Fer una funció anomenada `Estatística` que calculi i retorna alguns estadístics (màxim, mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre es incorrecta (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió es correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

af un programa complet (declaracions globals i funció main() ) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribucio per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaua (int taua[], int nElements);
```

que permet inicialitzar a zero una taula d'enteros de mida nElements. Utilitzeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les options.
7. Llegir l'opció escollida.
8. Si l'opció escollida és 1:
  - 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
  - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"
9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodí de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
10. Si l'opció escollida és 3:
  - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
  - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

Exercici 7 (0.3 punts)

Declarar un nou tipus de dades, TIOT, com un registre amb els camps: `Ident_lot` (valor enter), `Num_mitja_betais` (valor real), `Codi_venda` (valor enter) i `Origen_lot` (cadena de 50 caràcters).

Fer un procediment anomenat `InicialitzarIOT` que rebi com a paràmetre un registre del tipus TIOT i inicialitzi els seus camps amb valor zero als camps numèrics, i valor "" al camp de tipus cadena de caràcters.

Exercici 8 (1 punt)

Fer una funció anomenada `BuscarPrimerImparell` que rebi com a paràmetres un vector d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

**Exercici 8 (1 punt)**

**Ejercicio 8 (1 punto)**  
Fer un procedimiento anomenat `DobleMatriu` que doble el valor de tots els elements d'una matriu d'enters de  $n$  files i 5 columnes. El procedimiento recibirá como parámetros: una matriu d'enters de 5 columnas, el número de files de la matriu, i el número de columnas de la matriu.

1532906

X10<sup>6</sup>

min = v[0]

index min = 0

for (int i = 1; i < column; i++) {

if (v[i] < min)

min = v[i]

index min = i

< ist kein

ausreihen nötig std;

#define NUMDIREKTS 135

#define NUMBEVOTS 135

int main()

int Varts

int Resultat

int i = 0;

MenueFunktion()

cin << Opvar

1.  $\text{turn}$

else

return 0;

if  $\text{total} <$

total +

200

++x

0 if  $(x) \wedge f_1$

while ( $x < \text{num} \geq 0$ ):  $\text{turn}$

book &  $\text{total} = \text{total}$

$x = x - 1$

++x

$x = x - 1$  for

$f_1$

else

$s \neq \text{spur}$

$f_2 \text{ when } s \leq 50$

# Exàmens d'Informatica (TU38Ub)

Curs 2019-2020

Examen Parcial (11 de Novembre de 2019)

Nom estudiant: Aleix Calafell Araujo

NIU: 5925240

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessaries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherentia entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenui a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot es vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric (10,11,12,13) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una per als identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu nombre mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

## Exercici 1 (1 punt)

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats (10, 11, 12 ó 13). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$29 \times 3 = 87$$

$$87 - 12 = \boxed{75}$$

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)

La funció comprovarà que la posició on es volen guardar les dades sigui un index correcte (no inferior a zero, ni superior al darrer index vàlid de les taules). Si la posició és incorrecta, la funció retornara -1 i no es farà res més.

Si la posició és correcta, es demandarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardarà a la posició indicada de les taules corresponents.

En els dos casos, si s'introduceix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introdueixin els valors correctament.

Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzant la funció CodiVendaLot de l'ex. 1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

$$15 \times 2 = 30$$

$$30 + 5 = 35$$

$$\text{Result} = 35$$

### Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodi` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer paràmetre), i una taula on es guardarà el número de lots de cada codi de venda en una posició diferent.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estàrà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

### Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorna alguns estadístics (màxim, mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula

El valor de retorn de la funció serà -1 si la dimensió entrada com a segon paràmetre és incorrecta (més petita que 1); per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

### Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodi` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introdueixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador, número mitjà de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p.e.: "ID: 1234 num mitja de pètals: 32.6").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codí de venda inexistent" i es sortirà.

- per un programa complet (declaracions globals i funció main() ) que segueixi els següents passos:
1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
  2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
  3. Declarar (on correspongui) una estructura de dades anomenada distribucio per a guardar el número de lots que hi ha de cada codi de venda.
  4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula (int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.
  5. Utilitzar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.
  6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.
  7. Llegir l'opció escollida.
  8. Si l'opció escollida és 1:
    - 8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.
    - 8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"
  9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodi de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.
  10. Si l'opció escollida és 3:
    - 10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.
    - 10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.
  11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".
  12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

**Exercici 7 (0,5 punts)**

Declarar un nou tipus de dades, TLot, com un registre amb els camps: Ident\_lot (valor enter), Num\_mitja\_petal (valor real), Codi\_venda (valor enter) i Origen\_lot (cadena de 50 caràcters).

Fer un procediment anomenat InicialitzarLot que rebi com a paràmetre un registre del tipus TLot i inicialitzi els seus camps amb valor zero als camps numèrics, i valor "—" al camp de cadena de caràcters.

**Exercici 9 (1 punt)**

Fer una funció anomenada BuscarPrimerImparell que rebi com a paràmetres un vector d'enters i un valor enter amb el numero d'elements del vector. La funció ha de calcular i retornar en quin índex del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

**Exercici 8 (1 punt)**

Fer un procediment anomenat DobleMatriu que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

Important: Recordeu que cal donar les millors solucions possibles en cada exercici. A més de funcionar correctament, el codi ha de estar ben programat (codi clar, amb les instruccions més adients, sense operacions ni variables innecessàries, etc.)

Les funcions i procediments de les preguntes de la 1 a la 6 formen part d'un únic programa i, per tant, ha d'haver coherentia entre les seves definicions, i la forma en què s'utilitzen en altres preguntes. El context del programa a desenvolupar, el tenui a continuació:

BeautyFlors, una empresa dedicada al cultiu de flors, vol emmagatzemar dades i fer estadístiques sobre la seva producció de roses per tal de millorar la qualitat de la seva producció. La producció de roses es distribueix a partir de lots que venen identificats per un número (enter), i per a cada lot és vol guardar el nombre mitjà de pètals del lot i un codi de venda numèric ([0,11,12,13]) que indica el preu de venda del lot.

Aquesta informació es guardarà en tres taules (arrays) (una per als identificadors, una per les mesures del número de pètals i una pel codi de venda). La informació d'un mateix lot es guarda en la mateixa posició de les tres taules, és a dir, l'identificador d'un lot, el seu número mitjà de pètals i el seu codi de venda es guarden al mateix índex de les taules corresponents.

En total, l'empresa vol guardar les dades de 150 lots de flors.

**Exercici 1 (1 punt)**

Fer una funció anomenada CodiVendaLot per a classificar un lot en un dels codis de venda marcats ([0, 11, 12 ó 13]). La funció rebrà el valor del número mitjà de pètals del lot i retornarà el codi numèric de venda del lot segons el criteri següent:

- codi 10: si el número mitjà de pètals és inferior a 15
- codi 11: si el número mitjà de pètals és superior o igual a 15 cms i inferior a 25
- codi 12: si el número mitjà de pètals és superior o igual a 25 cms i inferior a 35
- codi 13: si el número mitjà de pètals és superior o igual a 35

NOTA: Considereu que els valors d'entrada són positius (no cal fer la comprovació).

$$352 - 121 = 231$$
$$231 + 43 = \underline{\underline{274}}$$

$$94 - 50 = 44$$

$$44 \times 2 = 88$$

$$\text{Resultat} : 88$$

Fer una funció anomenada EntrarDadesLot que demanarà les dades d'un lot de roses i les guardarà en les taules corresponents. La funció rebrà com a paràmetres:

- les tres taules amb els identificadors, els números de pètals mitjans i els codis de preu
- un enter amb la dimensió de les taules anteriors (un sol valor)
- la posició de la taula on es volen guardar les dades que s'introduixin

La funció comprovarà que la posició on es volen guardar les dades sigui un índex correcte (no inferior a zero, ni superior al darrer índex vàlid de les taules). Si la posició és incorrecta, la funció retornara -1 i no es farà res més.

Si la posició és correcta, es demanarà a l'usuari que introduixi el valor de l'identificador del lot i el número de pètals mitjà del lot de roses, i es guardarán a la posició indicada de les taules corresponents.

En els dos casos, si s'introduceix un valor que no és positiu, es mostrarà el missatge "Error: Valor incorrecte" i es tornarà a demanar el valor. Aquest procés es repetirà fins que s'introduceixin els valors correctament. Un cop introduïdes correctament les dades anteriors, es calcularà el codi de venda del lot (utilitzant la funció CodiVendaLot de l'ex. 1) i es guardarà a la taula corresponent. Si s'han introduït totes les dades correctament, la funció retornarà 0.

### Exercici 3 (1 punt)

Fer un procediment anomenat `DistribucioPerCodis` que calculi quants lots hi ha de cada codi de venda i ho guardi en una taula (array). El procediment rebrà com a paràmetres: una taula amb els codis de venda dels lots, la dimensió de la taula anterior (la passada com a primer parametre), i una taula on es guardarà el número de lots de cada codi de venda en una posició different.

NOTA: La taula on es guardarà el número de lots de cada codi de venda (tercer paràmetre) estarà inicialitzada a 0 (no cal inicialitzar-la en aquest procediment).

### Exercici 5 (1 punt)

Fer una funció anomenada `Estadistica` que calculi i retorni alguns estadístics (màxim, mínim) calculats sobre una taula de valors reals. La funció rebrà com a paràmetres:

- una taula de valors reals
- un enter amb la dimensió de la taula anterior (la taula passada com a primer paràmetre)
- dos variables reals per a guardar el màxim i el mínim dels valors de la taula (més petita que 1) i per tant no es poden calcular els estadístics. Si la dimensió és correcta i es poden calcular correctament el màxim i el mínim, la funció retornarà 0.

### Exercici 4 (1 punt)

Fer un procediment anomenat `ConsultaPerCodí` que permeti visualitzar les dades de tots els lots amb un determinat codi de venda. El procediment rebrà com a paràmetres:

- les tres taules amb els identificadors, els números mitjans de pètals i els codis de venda
- un enter amb la dimensió de les taules anteriors (un sol valor)

El procediment demanarà a l'usuari que introueixi el codi de venda que vol consultar. A continuació, el procediment mostrarà per pantalla les dades (identificador número mitja de pètals) dels lots que tinguin el codi de venda indicat per l'usuari. Les dades d'un mateix lot es mostraran en una mateixa línia (p. ex. "ID: 1234 num mitja de petals: 32.6").

Si s'introdueix un codi de venda que no és vàlid (codis vàlids: 10, 11, 12, 13), es mostrarà el missatge "Error: Codí de venda inexistent" i es sortirà.

el un programa complet (declaracions globals i funció main() ) que segueixi els següents passos:

1. Declarar (on correspongui) les constants necessàries per tal que el programa sigui fàcilment modificable.
2. Declarar (on correspongui) tres estructures de dades anomenades identificadors, mesures i codis per a guardar els identificadors dels lots de roses a estudiar, els valors dels seus números mitjans de pètals i els seus codis de venda, respectivament.
3. Declarar (on correspongui) una estructura de dades anomenada distribució per a guardar el número de lots que hi ha de cada codi de venda.
4. Suposeu que teniu implementat un procediment definit com:

```
void InicialitzarTaula(int taula[], int nElements);
```

que permet inicialitzar a zero una taula d'enters de mida nElements. Utilizeu (NO IMPLEMENTEU) aquest procediment per a inicialitzar a zero la taula distribucio.

5. Utilizar el procediment EntrarDadesLot de l'ex. 2, per a demanar les dades de TOTS els lots i guardar-les en les estructures de dades identificadors, mesures i codis.

6. Mostrar un missatge indicant que té 4 opcions disponibles (1-Distribució per codis de venda, 2- Consulta de lots per codi de venda, 3-Estadístics, 4-Sortir) i demanar a l'usuari que introdueixi una de les opcions.

7. Llegir l'opció escollida.

8. Si l'opció escollida és 1:

8.1. Cridar al procediment DistribucioPerCodis de l'ex. 3 per a calcular quants lots hi ha de cada codi de venda.

8.2. Mostrar per pantalla, quants lots hi ha de cada codi de venda amb missatges de l'estil: "Codi 10: 12 lots"

9. Si l'opció escollida és 2, cridar al procediment ConsultaPerCodí de l'ex. 4 per a visualitzar per pantalla les dades de tots els lots d'un codi de venda concret.

10. Si l'opció escollida és 3:

10.1. Cridar al procediment Estadistica de l'ex. 5 per a calcular el màxim i el mínim de les mesures (números mitjans de pètals) dels lots de roses.

10.2. Si la funció retorna un error, mostrar el missatge "Error: No s'han pogut calcular els estadístics". En cas contrari, mostrar per pantalla els valors del màxim i el mínim.

11. Si l'opció escollida no és cap de les 4 disponibles, mostrar el missatge "Opció incorrecta".

12. Repetir els passos del 6 al 11, fins que l'usuari seleccioni l'opció de sortir.

**Exercici 7 (2 punts)**  
Declarar un nou tipus de dades, TIOT, com un registre amb els camps: Ident\_lot (valor enter), Num\_mitja\_petals (valor real), Codi\_venda (valor enter) i Origen\_lot (cadena de 50 caràcters).

Fer un procediment anomenat InicialitzarLot que rebi com a paràmetre un registre del tipus TIOT i inicialitzi els seus camps amb valor zero als camps numèrics, i valor "" al camp de tipus cadena de caràcters.

**Exercici 8 (1 punt)**  
Fer una funció anomenada BuscarPrimerImparell que rebi com a paràmetres un vec d'enters i un valor enter amb el número d'elements del vector. La funció ha de calcular i retornar en quin index del vector hi ha el primer element ímparell. Si el vector no conté cap valor ímparell, retornarà -1.

**Exercici 8 (1 punt)**

Fer un procediment anomenat DobleMatriu que doble el valor de tots els elements d'una matriu d'enters de n files i 5 columnes. El procediment rebrà com a paràmetres: una matriu d'enters de 5 columnes, el número de files de la matriu, i el número de columnes de la matriu.

```

#include <iostream>
#include <define Diputats 135>
using namespace std;

int main()
{
 int Vots[Diputats];
 int Resultats[Diputats];
 int iden;
 int opcio; int parlamentari; int rot;
 int comp; int i; int sum=0; int puet, neli;

 InicialitzarArray(Vots, Diputats, 0);
 do {
 MenúPrincipal();
 cin >> opcio;
 switch (opcio) {
 case 1: cout << "Entri el seu identificador";
 cin >> iden;
 if (Whiteff(iden, Diputats) && (iden < 1)) {
 vot = votar(1, Diputats);
 Vots[iden] = vot;
 break;
 }
 case 2: comp = NoZeroArray(Vots, Diputats);
 if (comp == 1)
 InicialitzarArray(Resultats, Diputats, 0);
 Escratinif(Vots, Diputats, Resultats);
 for (i=0; i < Diputats; i++) {
 cout << "El diputat " << i << " té " << Resultats[i] <<
 " vots " << endl;
 }
 else
 cout << "Encara no es pot fer l'escrutini" << endl;
 break;
 }
 } while (opcio != 0);
}

```

case 3:

for(i=0; i<Departments; i++)

sum = sum + Results[i];

if (sum == 285)

Recall = IndexMaxArry(Results, Departments);  
Recall = IndexZeroArry(Results, Departments);

cout <<

else

cout <<  
break;

case 4:

break;

default:

cout <<