

Tema 5a: Taules unidimensionals

Problemes marcats en groc → Problemes avaluables

La resta de problemes són de dificultat intermèdia entre els avaluables. Si trobeu dificultats en passar d'un problema avaluable al següent, feu algun/s dels problemes no marcats com avaluables per a tenir un aprenentatge més progressiu.

Problemes marcats en blau → Problemes opcionals (no avaluables) però amb VPL disponible a Caronte per a auto-avaluació.

Problema 1

Fer un programa que declari un array d'enters de 10 posicions. Quan es declari el vector (= array) s'ha d'inicialitzar cada posició consecutiva amb els següents valors: 10, 9, 8, 7, 6, 5, 4, 3, 2, 1. Un cop declarat l'array, el programa ha d'imprimir per pantalla els elements de l'array amb els índexs 4 i 9 deixant un espai entre els dos.

Problema 2

Fes un programa que declari un array d'enters de 10 posicions. A continuació el programa ha de demanar a l'usuari que introdueixi un valor enter i ha d'inicialitzar tots els elements de l'array amb aquest valor. Per últim, el programa ha d'imprimir per pantalla tots els valors dels elements del vector (cada valor en una línia).

Problema 3

Fer un programa que declari un array d'enters de 10 posicions per a guardar les notes dels estudiants d'una classe. A continuació, el programa ha de demanar a l'usuari que introdueixi per teclat les notes dels estudiants i les guardi al vector. Si s'introdueix una nota incorrecta (menor que 0 o més gran que 10), s'ha de mostrar el missatge "Error: nota no valida" i s'ha de tornar a demanar a l'usuari que introdueixi la nota fins que la introdueixi correctament.

Un cop introduïdes correctament totes les notes, el programa ha de calcular el nombre d'aprovats i mostrar per pantalla el missatge: "Numero d'alumnes aprovats: NN" on NN serà el nombre d'aprovats calculat pel programa.

Problema 4

Fer un programa que demani a l'usuari que introdueixi valors per omplir un array d'enters de dimensió 10. L'usuari haurà d'anar introduint valors fins el màxim del vector o, si vol acabar anticipadament, fins que introdueixi un zero.

En acabar la introducció de valors, el programa mostrarà el missatge "Hem arribat al final de l'array" si l'usuari ha omplert tots els valors de l'array. En canvi, si no ha omplert tots els valors perquè ha acabat anticipadament introduint un zero, el programa mostrarà el missatge "Hem introduït un zero".

Problema 5

Fer un programa que donat un vector de 10 posicions digui si aquest està ordenat en ordre creixent o no. Si el vector està ordenat, el programa mostrarà el missatge "L'array esta ordenat" i en cas contrari, mostrarà el missatge "L'array no esta ordenat".

Problema 6

Fer tres procediments per a treballar amb vectors:

- LlegirVector: procediment per a omplir un vectors amb valors entrats pel teclat.
- EscriuVector: procediment per a escriure per pantalla el contingut d'un vector.
- SumaVectors: procediment per a sumar dos vectors i guardar el resultat en un tercer vector.

Com a paràmetres, aquests procediments rebran els vectors que necessitin i la seva mida per a que siguin procediments genèrics.

A continuació, fer un programa principal que utilitzi aquests procediments per a llegir dos vectors de 5 posicions, calcular la seva suma i mostrar el resultat per pantalla.

Problema 7

Fer un programa en el que es declarin quatre vectors d'enters (v1, v2, v3 i v4) de 10 posicions cadascun. Inicialitzeu el primer vector en el moment de la declaració posant a cada posició el valor del seu índex.

Després, inicialitzeu els altres vectors utilitzant bucles de la següent manera:

- a) A cada posició de v2 hi col·loqueu el doble del seu índex.
- b) A cada posició de v3 hi col·loqueu un enter entrat per l'usuari (se li demana a l'usuari un valor per cada posició).
- c) El vector v4 ha de ser la suma (element a element) dels altres 3 vectors.

Una vegada inicialitzats els vectors, el programa els ha de mostrar per pantalla per a comprovar que ho heu fet correctament. Els vectors han d'aparèixer un a cada línia amb els seus elements separats per un espai i amb l'encapçalament "Vector X: " on X serà 1, 2, 3 o 4.

Problema 8

Fer un programa en el que es declari un vector de 12 posicions i es demani a l'usuari que l'ompli amb valors entrats per teclat. Un cop l'usuari ha introduït tots els valors, el programa ha de mostrar per pantalla els valors del vector (separats per un espai) per a comprovar que s'han guardat correctament. A continuació, el programa ha de substituir tots els valors negatius del vector per un zero. Finalment, el programa ha de mostrar per pantalla els valors del vector per a comprovar que l'eliminació dels negatius s'ha fet correctament.

Quan mostreu els vectors per pantalla, imprimeix un vector en cada línia i amb cada element separat per un espai del valor anterior. Al davant del vector llegit, poseu l'encapçalament "Entrada: " i al davant del vector resultant després d'eliminar els negatius "Sortida: ".

Problema 9

Una empresa de fabricació de planxes de metall produeix diàriament 20 planxes i al final del dia volen poder saber el percentatge de planxes produïdes correctament

Fer un programa que permeti emmagatzemar les dades de producció d'un dia en un vector. El programa ha d'anar mostrant un missatge de l'estil "Introdueix el gruix de la planxa N: " per a demanar a l'operari que introdueixi el gruix en mil·límetres de cadascuna de les 20 planxes produïdes i aquesta informació s'ha de guardar en un vector. Si l'operari introdueix un valor igual o menor que zero, se li ha de tornar a demanar que introdueixi el valor.

A continuació, el programa ha de calcular el percentatge de planxes que tenen un gruix entre 5mm i 10mm (ambdós inclosos) que són els límits marcats per l'empresa com a valors correctes de les planxes.

Problema 10

Fer un programa que permeti guardar en una estructura de dades adient la informació sobre el mes en el que han començat a caminar els nens i nenes d'una escola bressol. El programa demanarà a l'usuari que s'introdueixi la informació amb el missatge "Introdueix els mesos que tenia el/la nen/a N quan va caminar per primer cop: " i llegirà el valor de cadascun/a dels 15 nens/es de l'escola de forma consecutiva. El programa haurà de comprovar que la mesura sigui vàlida (entre 9 i 18, ambdós inclosos). Si la mesura no està entre aquests dos valors es donarà el següent missatge d'error "Error: Valor no valid" i es tornarà a demanar el valor del nen/a N.

Una vegada introduïdes totes les dades, el programa ha de calcular la mitjana de les dades i informar de l'edat mitjana en què els nens i nenes de l'escola van començar a caminar: "Mitjana d'edat en caminar: XX mesos" on XX serà la mitjana de l'edat en mesos en que van començar a caminar els infants de l'escola.

Nota: En el cas d'haver caminat abans dels 9 mesos o després dels 18 mesos l'usuari introduirà 9 ó 18 respectivament.

Problema 11

Fer un programa en el que es declari un vector de 10 posicions i es demani a l'usuari que l'ompli amb valors enters positius entrats per teclat. Cal controlar que els valors introduïts per l'usuari siguin positius. Si l'usuari introdueix algun valor que no ho és, el programa ha de mostrar un missatge d'error i tornar a demanar que introdueixi el valor fins que ho faci correctament.

A continuació, el programa ha de calcular i mostrar per pantalla quin és el màxim del vector i en quina posició es troba.

Problema 12

Fer un programa en el que es declari un vector de 10 posicions i es demani a l'usuari que l'ompli amb valors enters entrats per teclat. A continuació, el programa ha de calcular i mostrar per pantalla quin és el màxim del vector i en quin índex es troba amb el missatge: "El maxím del vector es MM i es troba a l'índex PP" on MM és el valor del màxim del vector i PP és el seu índex dins del vector.

Problema 13

Feu un programa que emmagatzemi les edats dels socis d'un gimnàs amb la finalitat de fer-ne estadístiques. El programa ha de crear un vector amb tantes posicions com socis hi hagi al gimnàs (aquest valor haurà de ser una constant definida a l'inici del programa amb un define) i demanar totes les edats a l'usuari. Després ha de calcular i mostrar per pantalla les següents dades:

- a) Mitjana d'edat dels socis
- b) Edat màxima i mínima
- c) Número de persones que són majors d'edat

Problema 14

Fer un programa que llegeixi la temperatura mitjana dels dotze mesos de l'any i els guardi en una estructura de dades adient (les temperatures seran enters).

A continuació el programa ha de calcular i mostrar per pantalla quin mes (identificat amb un número) ha tingut la temperatura mínima i quin la màxima. El format del missatge de sortida serà el següent:

Mes amb temperatura mínima: MM

Mes amb temperatura màxima: NN

on MM i NN seran els números dels mesos (1=gener, 2=febrer, etc.) amb temperatura mínima i màxima respectivament.

Després, el programa ha de calcular i mostrar la temperatura mitjana anual amb el missatge "Temperatura mitjana: MM" on MM és la temperatura mitjana i, per últim, ha de dir quins mesos han tingut una temperatura inferior a la mitjana anual, quins una temperatura superior i quins una temperatura igual. El format dels missatges de sortida (un per cada mes) serà:

El mes MM ha tingut temperatura inferior a la mitjana anual.

El mes MM ha tingut temperatura superior a la mitjana anual.

El mes MM ha tingut temperatura igual a la mitjana anual.

Problema 15

Fer un programa que demani a l'usuari que introdueixi valors senars per omplir un vector de 9 posicions. Aquest pas el realitzarem amb un recorregut del vector que llegirà els valors de l'usuari sense comprovar si els valors són senars o no.

Una vegada el vector està ple amb els valors de l'usuari, el programa comprovarà si tots els valors del vector són senars. En aquest cas, el programa mostrarà el missatge "TOTS SON SENARS". En cas contrari, mostrarà el missatge "NO TOTS SON SENARS".

Important: El programa no ha de fer cap operació innecessària. Quan es comprova si tots els elements del vector són senars, si es detecta un nombre no senar, el programa no ha de seguir comprovant la resta d'elements.

Problema 16

Fer un programa que demani a l'usuari que introdueixi valors reals per omplir un vector de 20 posicions. Aquest pas el realitzarem amb un recorregut del vector que llegirà els valors de l'usuari sense fer cap comprovació.

A continuació, el programa buscarà a quina posició hi ha el primer valor superior a 100. Si en troba algun, mostrarà el missatge "El primer element superior a 100 es a la posició XX" on XX serà la posició on ha trobat l'element. Si no ha trobat cap element superior a 100, es mostrarà el missatge "No hi ha cap element del vector superior a 100".

Important: El programa no ha de fer cap operació innecessària. Quan es trobi el primer element superior a 100, ja no cal continuar buscant al vector.

Problema 17

Fer un programa que demani a l'usuari que introdueixi valors reals per omplir un array de 10 posicions. Aquest pas el realitzarem amb un recorregut de l'array que llegirà els valors de l'usuari sense fer cap comprovació.

A continuació, el programa ha d'anar sumant els valors de cadascuna de les posicions de l'array des de la primera posició fins que el valor acumulat sigui més gran que 25. Si

s'ha aconseguit acumular aquest valor, el programa mostrarà el missatge "A la posició PP la suma acumulada es superior a 25" on PP serà la posició de l'array on s'ha superat la suma acumulada de 25. Comptarem les posicions a partir de 1 (tingueu en compte que els índexs dels arrays van de 0 fins al nombre de posicions menys 1). Si la suma acumulada de tot l'array no supera el valor de 25, el programa mostrarà el missatge "La suma acumulada de l'array es inferior o igual a 25".

Important: El programa no ha de fer cap operació innecessària. Quan es trobi la posició on la suma acumulada és superior a 25, ja no cal continuar sumant la resta de posicions.

Problema 18

Fer un programa que demani a l'usuari que introdueixi valors enters per omplir un array de 12 posicions. Aquest pas el realitzarem amb un recorregut del vector que llegirà els valors de l'usuari sense fer cap comprovació.

A continuació, el programa ha de demanar a l'usuari un valor i ha de buscar aquest valor a l'array. Si el troba, el programa ha de mostrar el missatge "Valor trobat a la posició XX" on XX és la posició de l'array on s'ha trobat el valor introduït per l'usuari. Si el valor no es troba a l'array, el programa ha de mostrar el missatge "Valor no trobat".

Important: El programa no ha de fer cap operació innecessària. Si es trobi el valor a l'array, ja no cal continuar comprovant si el valor és a la resta de posicions.

Problema 19

Fer un programa que demani a l'usuari que introdueixi valors enters per omplir dos arrays de 6 posicions (primer un array i després l'altre). Aquest pas el realitzarem amb dos recorreguts dels arrays que llegiran els valors de l'usuari sense fer cap comprovació.

A continuació, el programa ha de comprovar si els dos arrays són iguals o no. Si són iguals, el programa mostrarà el missatge "IGUALS" i si són diferents mostrarà el missatge "DIFERENTS".

Important: El programa no ha de fer cap operació innecessària. En la comparació dels arrays, quan es trobi la primera posició diferent, no s'ha de continuar comprovant la resta de posicions.

Problema 20

Fer una petita llibreria d'operacions genèriques amb vectors d'enters. Per a fer-ho, heu de programar els procediments que teniu a continuació en un únic fitxer "vectors.cpp" i heu de fer el fitxer de capçalera corresponent "vectors.h":

Procediment **InicialitzarVector** que té tres paràmetres: un vector d'enters, el valor amb el que es vol inicialitzar el vector i el número d'elements del vector. Aquest procediment inicialitza cada posició del vector amb el valor passat com a segon paràmetre.

Procediment **LlegirVector** que té dos paràmetres: un vector d'enters i el número d'elements del vector. Aquest procediment demana a l'usuari que entri per teclat el valor de cada posició del vector i el guarda a la posició corresponent del vector.

Procediment **EscriureVector** que té dos paràmetres: un vector d'enters i el número d'elements del vector. Aquest procediment mostra per pantalla els valors del vector separats per un espai.

Algunes d'aquestes operacions (o molt similars) han estat fetes en problemes anteriors i podeu aprofitar el codi corresponent per a crear les funcions i procediments d'aquest problema.

Una vegada creada la llibreria d'operacions sobre vectors, feu un programa principal que utilitzi aquestes funcions i comproveu que funcionen correctament.

Problema 21

Continuar amb la llibreria d'operacions genèriques amb vectors d'enters afegint als fitxers “vectors.cpp” i “vectors.h” les funcions següents:

Funció **MitjanaVector** que té dos paràmetres: un vector d'enters i el número d'elements del vector, i retorna un valor real. Aquesta funció calcula la mitjana dels elements del vector i retorna el seu valor com un número real.

Funció **MaximVector** que té dos paràmetres: un vector d'enters i el número d'elements del vector, i retorna un valor enter. Aquesta funció busca el valor màxim dels elements del vector i retorna el seu índex com un número enter.

Funció **MinimVector** que té dos paràmetres: un vector d'enters i el número d'elements del vector, i retorna un valor enter. Aquesta funció busca el valor mínim dels elements del vector i retorna el seu índex com un número enter.

Funció **MinimVectorNoZero** que té dos paràmetres: un vector d'enters i el número d'elements del vector, i retorna un valor enter. Aquesta funció busca el valor mínim dels elements del vector sense tenir en compte els valors que són zero i retorna el seu índex com un número enter. Si tots els valors del vector són zero, retorna -1.

Algunes d'aquestes operacions (o molt similars) han estat fetes en problemes anteriors i podeu aprofitar el codi corresponent per a crear les funcions i procediments d'aquest problema.

Una vegada creada la llibreria d'operacions sobre vectors, feu un programa principal que utilitzi aquestes funcions i comproveu que funcionen correctament.

Problema 22

Implementar el problema 14 (estadístiques sobre les temperatures mitjanes dels 12 mesos de l'any) seguint els principis de la programació modular, és a dir, posant cada tasca en una funció. Per a fer-ho, heu d'utilitzar les funcions i procediments dels problemes 20 i 21 (vectors.cpp i vectors.h), i substituir el codi que fa alguna de les operacions de la llibreria per la crida a la funció o procediment corresponent.

Problema 23

Els estudiants de quart curs volen fer un programa per a gestionar la votació sobre la destinació del viatge de fi de carrera. Les opcions de vot s'identifiquen amb un nombre enter i són: 1-Cancun, 2-Cuba, 3-Creta, 4-Croàcia, 5-Canàries. El vot (un número del 1 al 5) de cada estudiant es guardarà en un vector d'enters amb la mida adequada. A quart curs hi ha 156 estudiants.

En primer lloc, feu un procediment anomenat **LlegirVots** que permeti introduir els vots de tots els estudiants a l'estructura de dades. El procediment rebrà el vector que guarda els vots com a paràmetre i anirà demanant a l'usuari que introdueixi els vots dels estudiants amb un missatge del tipus “Introdueix el vot de l'estudiant n”. El procediment haurà de comprovar que el vot introduït sigui vàlid (entre 1 i 5, ambdós inclosos). Si el

vot no està entre aquests dos valors es donarà el següent missatge d'error "Vot no vàlid" i es tornarà a demanar el vot de l'estudiant n.

A continuació, fer el programa principal on es declari un vector per a guardar els vots i es faci la crida al procediment LlegirVots. A continuació, s'ha de determinar quina és l'opció guanyadora buscant el màxim del vector de vots (utilitzeu les funcions fetes al problema 20) i s'ha de mostrar quina és aquesta opció per pantalla.

Problema 24

Fer un programa que llegeixi i guardi en una estructura de dades adient les temperatures màximes de cada dia durant un període de dues setmanes (14 dies) i permeti saber quina és la temperatura diària màxima que s'ha repetit més cops al llarg de tots els dies del període. Podem suposar que la temperatura mai serà inferior a -10° ni superior a 50° (valors enters).

El programa haurà de demanar a l'usuari la temperatura màxima de tots els dies i haurà de construir una taula (histograma de temperatures) en què per a cada possible valor de temperatura (de -10 a 50) es guardi quants dies han tingut aquella temperatura al llarg del període.

A continuació, el programa ha de calcular quina és la temperatura que s'ha repetit més cops al llarg dels 14 dies i quina és la temperatura que s'ha repetit menys vegades i mostrar-ho per pantalla amb el missatge: "Temperatura mes repetida: XX - Temperatura menys repetida: YY" on XX i YY seran dos valors enters entre -10 i 50 i correspondran a les temperatures amb valor més gran i més petit (diferent de zero) de l'histograma de temperatures. En el cas que hi hagi més d'una temperatura que siguin les més/menys repetides, agafarem la primera temperatura que trobem com a més/menys repetida.

Important: Utilitzeu les funcions dels problemes 20 i 21 per a fer les diferents operacions que cal fer en aquest problema.

Problema 25

Fer la funció Detectar1000 que rep com a paràmetres un array d'enters i la seva dimensió. La funció retorna un 1 si algun dels valors de l'array és més gran o igual a 1000 i un 0 si tots els valors de l'array són inferiors a 1000.

Fer també el programa principal que demani un array d'enters a l'usuari, cridi a la funció Detectar1000 i mostri un missatge amb el valor retornat per a comprovar que la funció funciona correctament.

Problema 26

Fer la funció OrdenatDecreixent que rep com a paràmetres un array d'enters i la seva dimensió. La funció retorna un 1 si l'array és una seqüència de nombres ordenada en ordre de decreixent i un 0 en cas contrari.

Fer també el programa principal que demani un array d'enters a l'usuari, cridi a la funció OrdenatDecreixent i mostri un missatge amb el valor retornat per a comprovar que la funció funciona correctament.

Problema 27

Fer la funció ParellsCreixent que rep com a paràmetres un array d'enters i la seva dimensió. La funció retorna un 1 si l'array és una seqüència creixent de nombres parells consecutius i un 0 en cas contrari.

Fer també el programa principal que demani un array d'enters a l'usuari, cridi a la funció ParellsCreixent i mostri un missatge amb el valor retornat per a comprovar que la funció funciona correctament.

Problema 28

Fer la funció **BuscarPosicio** que rep com a paràmetres un array d'enters ordenat en ordre decreixent, un valor enter i la dimensió de l'array. La funció busca en quina posició de l'array li correspondria anar al valor entrat com a segon paràmetre. És a dir, busca quina és la primera posició de l'array que conté un valor menor que el segon paràmetre. El valor de la posició on hauria d'anar el valor serà el valor de retorn de la funció. Si s'arriba al final de l'array i cap element és menor que el valor, la funció retornarà -1.

Després, fer un programa que demani a l'usuari que introdueixi un array d'enters de 5 posicions, on els nombres estiguin ordenats en ordre decreixent. A continuació, utilitzar la funció **OrdenatDecreixent** del problema 26 per a comprovar que realment l'array està correctament ordenat. Si és correcte, el programa demanarà a l'usuari que introdueixi un valor i farà la crida a la funció **BuscarPosicio**. Si la funció retorna -1 el programa mostrarà el missatge "Tots els valors del vector son mes grans que XX" on XX serà el valor introduït per l'usuari, i en cas contrari mostrarà el missatge "El valor ha d'anar a la posicio PP" on PP serà el valor retornat per **BuscarPosicio**. Si el vector no està ordenat en ordre decreixent es mostrarà el missatge "El vector no esta correctament ordenat".

Problema 29

Fer la funció **InversArray** que rep com a paràmetres dos arrays d'enters (els dos de la mateixa longitud) i la seva dimensió. La funció retorna un 1 si els dos arrays són el mateix un llegit d'esquerra a dreta i l'altre de dreta a esquerra i un 0 si són diferents.

Fer també el programa principal que demani un array d'enters a l'usuari, cridi a la funció **InversArray** i mostri un missatge amb el valor retornat per a comprovar que la funció funciona correctament.

Problema 30

Completar la llibreria d'operacions genèriques amb vectors d'enters feta al problema 20 amb les funcions i procediments següents:

Procediment **LlegirVectorStop** que té tres paràmetres: un vector d'enters, el número d'elements del vector i un valor de parada (stop). Aquest procediment demana a l'usuari que entri per teclat el valor de cada posició del vector i el guarda a la posició corresponent del vector. Si en alguna posició s'introdueix el valor de parada, s'acaba l'entrada de dades al vector.

Funció **PositiuVector** que té dos paràmetres: un vector d'enters i el número d'elements del vector, i retorna un valor enter. Aquesta funció compta i retorna el nombre de valors positius que hi ha al vector.

Funció **NegatiuVector** que té dos paràmetres: un vector d'enters i el número d'elements del vector, i retorna un valor enter. Aquesta funció compta i retorna el nombre de valors negatius que hi ha al vector.

Funció **StdevVector** que té dos paràmetres: un vector d'enters i el número d'elements del vector, i retorna un valor real. Aquesta funció calcula la desviació estàndard dels elements del vector i retorna el seu valor com un número real.

Problema 31

Fer un programa que llegeixi i guardi una sèrie de fins a 25 nombres enters acabada amb el 0. És a dir, s'han d'anar llegint nombres per teclat i parar quan s'introdueixi un 0 o bé ja s'hagin llegit 25 valors. Després de llegir i guardar la sèrie, s'ha de calcular i mostrar per pantalla el màxim i el mínim, el nombre de valors positius i de valors negatius de la sèrie, la mitjana i la desviació estàndard.

Per a fer aquest problema, utilitzeu les operacions de vectors programades en la llibreria feta als problemes 21 i 30.

Problema 32

Fer el procediment OrdenarVector que rep com a paràmetres un array d'enters i un enter amb la mida de l'array. El procediment ha d'ordenar els elements de l'array de més petit a més gran.

Fer també un programa que demani a l'usuari que introdueixi per teclat 8 nombres enters (que poden estar en qualsevol ordre) i els guardi en un array. A continuació utilitzar el procediment OrdenarVector per a ordenar de més petit a més gran l'array introduït per l'usuari i mostrar per pantalla l'array ordenat amb el format "Array ordenat: X Y Z ..." on X,Y,Z,etc. seran els valors del vector separats per un espai.

Per a fer aquest problema, busqueu informació sobre algorismes d'ordenació d'arrays (per exemple, el mètode de l'intercanvi, el mètode de la bombolla o el mètode d'inserció).

Problema 33

Fer el procediment EliminarElement que rep com a paràmetres un vector d'enters, un enter amb la mida del vector i un valor enter. Suposant que la taula està ordenada en ordre creixent, el procediment busca el valor passat com a tercer paràmetre en el vector i si el troba, l'elimina desplaçant la resta de valors que té a la dreta una posició cap a l'esquerra.

A continuació fer un programa que demani a l'usuari que introdueixi per teclat un vector de 10 elements i després cridi al procediment OrdenarVector per a ordenar de més petit a més gran el vector. Després, s'ha de demanar a l'usuari que introdueixi un valor enter i utilitzant el procediment EliminarElement eliminar el valor introduït per l'usuari del vector. Per últim, mostrar per pantalla el vector resultant.

Problema 34

Fer el procediment EliminarRepetits que rep com a paràmetre una taula unidimensional de N elements diferents de zero i un enter amb la mida del vector. El procediment ha d'eliminar tots els elements repetits de la taula. La taula resultant haurà de tenir 0 a les posicions ocupades inicialment pels elements repetits. La primera aparició de cada element repetit s'ha de mantenir a la taula final. Totes les operacions s'han de fer sobre la taula original.

Fer també el programa principal que demani un array d'enters a l'usuari, cridi al procediment EliminarRepetits i mostri el vector després d'eliminar els repetits per a comprovar que el procediment funciona correctament.