

Vectores, Táboas e Táboas Multidimensionais (Arrays)

Existen tipos de problemas nas que é preciso almacenar un número de valores indeterminado antes da execución do programa, polo que non é posible saber o número de variables que temos que utilizar, ou ben o número de variables sexa excesivo e o programa resultante pouco eficiente.

Exemplo:

Nun centro de ensino teñen un máximo de 40 alumnos por clase. Escribir un algoritmo que pregunte o número de alumnos e solicite o nome e a nota de cada un deles. Unha vez finalizado, nos amosará a nota media, o número de aprobados, o número de suspensos e o listado dos alumnos aprobados e o listado de alumnos suspensos coas súas notas correspondentes.

Como podemos ver, non é posible listar os alumnos con nota maior ou igual a 5 e os alumnos con nota menor a non ser que os teñamos almacenados en variables..... ¿pero cantas variables preciso? ¿40 para os nomes e 40 para as notas?. Non sei o número de alumnos no momento de escribir o programa, e sería realmente difícil procesar esa información.

Para estos casos se utiliza un tipo de variables especial que fan referencia a zonas de memoria que almacenan varios elementos de información, aos que podemos acceder mediante un ou varios índices. Si utilizan un so índice se denominan Vectores, si utilizan 2 ou mais Táboas ou Matrices, e xenericamente se coñecen como Arrays. Normalmente non é necesario utilizar Arrays de máis de dúas ou tres dimensións. O exercicio anterior poderíamos solucionalo do seguinte xeito:

Visualizar “Cantos alumnos? :”

Ler n_alumnos;

c=0 // Para a conta dos alumnos

c_aprobados=0 // Para a conta dos aprobados

c_suspensos=0 // Para a conta dos suspensos

suma=0 // Para levar a suma de todas as notas e poder calcular a media

Mentres (c<n_alumnos)

Visualizar “Nome :?”

Ler nome[c]; // Cando c vale 0, se almacenará en nome[0], cando valga 1 en nome[1]... etc.

Visualizar “Nota:?”

Ler nota[c]; // Cando c vale 0 se almacenará en nota[0], cando valga 1 en nome[1]... etc.

suma = suma + nota[c]

Se (nota[c] >= 5) // Aprobado

aprobados[c_aprobados]=c; // Gardo o índice do alumno aprobado

c_aprobados=c_aprobados+1 // Teño un alumno aprobado máis

Se-Non // Suspenso

suspensos[c_suspensos]=c; // Gardo o índice do alumno suspenso

c_suspensos=c_suspensos+1 // Teño un alumno suspenso máis

Fin-Se

c = c + 1 // Contamos un alumno máis

Fin-Mentres

Visualizar “A Nota Media é de “ suma/n_alumnos

Visualizar “Aprobados: “ c_aprobados “, Suspensos: “ c_suspensos

Visualizar “Lista de Aprobados: “

c=0 // Conta dos aprobados

Mentres (c < c_aprobados)

// en aprobados[c] teño o índice do alumno aprobado

Visualizar “Nome: “ nome[aprobados[c]] “ --- Nota: “ nota[aprobados[c]]

c = c + 1 // Seguinte aprobado

Fin-Mentres

Visualizar “Lista de Suspensos:”

c=0 *// Conta dos suspensos*

Mentres(c < c_suspensos)

// en suspensos[c] teño o índice do alumno aprobado

Visualizar “Nome: “ nome[suspensos[c]] “ --- Nota: “ nota[suspensos[c]]

c = c + 1 *// Seguinte suspenso*

Fin-Mentres