MÉTODOS MATEMÁTICOS (PARTE INFORMÁTICA)

PRIMER EXAMEN PARCIAL 2004

Ejercicio 1. A la vista del siguiente programa C, se desea que el alumno diga la **salida** que se produciría si el programa se ejecuta y se siguen estrictamente las órdenes que vayan apareciendo en la pantalla.

```
#include <stdio.h>
void main()
char cadena[21], resultado[21];
char key[27]="ABCDEFGHIJKLMNOPQRSTUVWXYZ";
int i,j,c;
printf("Introduce tu nombre y apellidos EN LETRAS MAYUSCULAS,\n");
printf("hasta un máximo de 20 caracteres\n");
for(i=0;(cadena[i]=getchar())!='\n';i++);
for(i=0;i<20;i++)
       if(cadena[i]==' ')
              resultado[i]=cadena[i];
              continue;
       for(j=0;j<26;j++)
              if(cadena[i]==key[j])
                     if((j+3) \le 25) c = j+3;
                     else (c=j-25+2);
                     resultado[i]=key[c];
              }
       }
printf("%s\n",resultado);
```

Para facilitar la corrección se habilitan los siguientes espacios para escribir los datos introducidos y los resultados obtenidos. Se ruega que los rellenen con la mayor claridad posible

Datos introducidos:

Res	Resultado obtenido:																

Primer parcial 2004. Solución al ejercicio 1

Cada letra del alfabeto se sustituye por la tercera a partir de ella, volviendo a empezar por el principio cuando esa tercera letra pasa de la Z. Los espacios en blanco los deja tal cual.

Por ejemplo, para esta entrada:

D .	• .	1	
Datos	intro	duc	1d0e1
Daios	muo	uuc	iuos.

P	Е	D	R	О		S	A	N	C	Н	Е	Z		M	A	R	Τ	Ι	N
Dag	December de la characida.																		
Kes	Resultado obtenido:																		
S	Н	G	U	R		V	D	Q	F	K	Н	C		P	D	U	W	L	Q

Ejercicio 2

Dada una matriz a[][] real, llamaremos: S al conjunto de los coeficientes de la matriz situados encima de la diagonal principal, D a los coeficientes de la diagonal principal e I a los coeficientes situados debajo de la diagonal principal.

Se pide:

Escribir un programa en lenguaje C que lea como datos:

- 1) La dimensión (n) de una matriz cuadrada (a). (n <= 100).
- 2) Los coeficientes de la matriz a.

Como resultado, debe imprimir en pantalla:

El **valor máximo** de los elementos del conjunto S (**ms**), del conjunto D (**md**) y del conjunto I (**mi**), respectivamente.

(Se deben utilizar obligatoriamente los nombres de las variables que están indicados entre paréntesis).

Primer parcial 2004. Solución al ejercicio 2

```
#include <stdio.h>
void main( )
int i,j,n;
float a[100][100],ms,md,mi;
printf("Introduce el valor de n\n");
scanf("%d",&n);
printf("Introduce los coeficientes de la matriz a");
for(i=0;i< n;i++)
for(j=0;j< n;j++)
scanf("%f", &a[i][j]);
/* Cálculo de ms */
ms=a[0][1];
for(i=0;i<n-1;i++)
for(j=i+1;j< n;j++)
if(a[i][j]>ms) ms=a[i][j];
/* Cálculo de md */
md=a[0][0];
for(i=1;i<n;i++)
if(a[i][i]>md) md=a[i][i];
/* Cálculo de mi */
mi=a[1][0];
for(j=0;j< n-1;j++)
for(i=j+1;i < n;i++)
if(a[i][j]>mi) mi=a[i][j];
printf(" ms=%f md= %f mi=%f \n", ms,md,mi);
```

Ejercicio 3

El programa C incompleto de la parte posterior de la hoja, se encuentra descrito a continuación:

- Los datos de entrada son las coordenadas de 20 puntos en el plano (distintos del origen y ejes) y el radio de un círculo centrado en el origen. (datos de tipo real)
- Se calcula el cuadrante al que pertenece cada punto (1, 2, 3 ó 4) y si es interior (estricto) al círculo o no
- En un vector estructura (v) se almacenan tanto las coordenadas de los puntos, como el número del cuadrante y la información de si es interior o no.
- La salida a pantalla debe ser:

El punto está en el cuadrante xxxx y zzzzzzzzzzzzzz al círculo (Donde xxxx es el entero 1, 2, 3 ó 4; y zzzzzzzzzzzzzzzzzzz es la cadena "es interior" o "no es interior" según el caso.)

Esta frase se escribe sólo para uno de los puntos: el punto cuyo número de orden introduce el usuario y se lee en la variable n.

- La escritura de la cadena "es interior" o "no es interior" se realizará mediante el operador condicional.

Se pide **completar** las líneas donde aparecen puntos suspensivos con el código adecuado, para que el programa funcione correctamente y de acuerdo a la descripción hecha antes.

(la longitud de la línea de puntos suspensivos no tiene relación directa con la longitud del código a completar)

```
#include <stdio.h>
void main()
     int i,x,n;
     float radio;
     struct puntos
          float p.....;
          int cua:
          int interior;
     ......
     /* lectura de datos*/
     for (i=0;i<20;i++)
          printf("Introduce las coordenadas del punto %d\n",i+1);
          .....
     printf("Introduce el radio del círculo\n");
     .....
     /* Determinación del cuadrante*/
     for(i=0;i<20;i++)
     {
          if(v[i].p[0]>0 && v[i].p[1]>0) v[i].cua=1;
          else if (v[i].p[0]>0 && v[i].p[1]<0)v[i].cua=4;
          else if (v[i].p[0]<0 && v[i].p[1]>0)v[i].cua=2;
          .....v[i].cua=3;
     }
     /* Posición respecto al círculo*/
     for(i=0;i<20;i++)
          x=.....
          switch (x)
          {
          case 0:
               v[i].interior=0;
          case 1:
               v[i].interior=1;
     printf("Introduce el número de orden del círculo del que quieres saber la
información\n");
     scanf("%d",&n);
     printf("El punto está en el cuadrante %d y %s al círculo",
.....);
```

Primer parcial 2004. Solución al ejercicio 3

```
#include <stdio.h>
void main()
       int i,x,n;
       float radio;
       struct puntos
              float p[2];
              int cua;
              int interior;
       v[20];
       /* lectura de datos*/
       for (i=0;i<20;i++)
              printf("Introduce las coordenadas del punto %d\n",i+1);
              scanf("%f %f",&v[i].p[0],&v[i].p[1]);
       printf("Introduce el radio del círculo\n");
       scanf("%f",&radio);
       /* Determinación del cuadrante*/
       for(i=0;i<20;i++)
       {
              if(v[i].p[0]>0 && v[i].p[1]>0) v[i].cua=1;
              else if (v[i].p[0]>0 && v[i].p[1]<0)v[i].cua=4;
              else if (v[i].p[0]<0 && v[i].p[1]>0)v[i].cua=2;
              else v[i].cua=3;
       }
       /* Posición respecto al círculo*/
       for(i=0;i<20;i++)
               x=v[i].p[0]*v[i].p[0]+v[i].p[1]*v[i].p[1]-radio*radio<0;
               switch (x)
              case 0:
                      v[i].interior=0;
                      break;
              case 1:
                      v[i].interior=1;
       printf("Introduce el número de orden del círculo del que quieres saber la
información\n");
       scanf("%d",&n);
       printf("El punto está en el cuadrante %d y %s al círculo",
                       v[n-1].cua,v[n-1].interior?"es interior":"no es interior");
}
```