

Lea atentamente estas instrucciones antes de comenzar:

- Responda a cada una de las cuestiones prácticas.
- Es necesario aprobar este examen (5 puntos) para aprobar la asignatura.
- No es suficiente este examen para aprobar la asignatura: se deben aprobar los dos trabajos de la parte práctica.

Sistemas de álgebra computacional: Maxima

1. (1 punto) ¿Para qué operación se usa el símbolo “:” en Maxima?

Respuesta: El símbolo ":" se emplea para realizar asignaciones.

2. (1 punto) ¿Qué caracteres empleamos en Maxima para definir una lista?

Respuesta: Se emplean "[", "]" y ",". Los dos primeros indican el comienzo y final de lista, respectivamente, mientras que la coma se emplea para separar los elementos de la lista.

3. (1 punto) ¿Para qué se emplea el símbolo “\$” en Maxima?

Respuesta: Se emplea para indicar el final de una instrucción de la que no deseamos obtener el output correspondiente escrito por pantalla.

4. (1 punto) ¿Para qué se emplea el símbolo “%” en Maxima?

Respuesta: El símbolo % aislado es un carácter reservado asignado al output de la última instrucción ejecutada, independientemente de que dicho output haya sido visualizado o no.

El símbolo % unido a otros caracteres se emplea para designar diversas constante matemáticas. Por ejemplo pi, e, i no llevan asociado ningún valor en Maxima, mientras que %pi, %e, %i son, respectivamente, el número pi, el número e y la unidad imaginaria.

Programación en C

1. (4 puntos) El modelo de Ising consiste en un conjunto de N partículas cuánticas (“espines”) cuyo número cuántico s puede tomar sólo los valores $+\frac{1}{2}$ o $-\frac{1}{2}$. Los espines interactúan entre sí, de forma que la energía del sistema aumenta cuando un par de espines en interacción son paralelos y disminuye cuando son antiparalelos; usualmente se dice que cada par de espines s_i y s_j suma a la energía del sistema un valor $s_i s_j$. Los pares de espines en interacción pueden ser todos los pares que se pueden formar (en total, $N(N - 1)/2$ pares), o sólo los pares consecutivos en la línea de espines (N pares).

Escríbase una función

```
float ising_U(int N, float s [], int local);
```

que, al pasarle el número de espines, el array con los valores de los espines y un valor que indique si la interacción es global (0: todos los pares posibles) o local (1: sólo los vecinos próximos) calcule la energía del modelo de Ising.

```

Solución: float ising_U(int N, float s[], int local) {
    int i, j;
    float egy=0.0;
    if( local ) {
        for(i=1; i < N; i++)
            egy+=s[i]*s[i-1];
        egy+=s[0]*s[N-1];
    } else {
        for(i=0; i < N; i++)
            for(j=0; j < i; j++)
                egy+=s[i]*s[j];
    }
    return egy;
}

```

2. (2 puntos) Supongamos que tenemos un archivo de datos "datos.dat" con N pares de puntos que representan el comportamiento de una función en un intervalo dado. Escribir un programa que lea esos puntos y obtenga los valores de x en los que la función alcanza el valor máximo y el valor mínimo dentro del intervalo.

```

Solución: #include <stdio.h>
#include <stdlib.h>

```

```

int main(int argc, char** argv) {
    int i, n, npts;
    double xx, yy, ymin, ymax, xmin, xmax;
    FILE *fin;
    fin=fopen("datos.dat", "r");
    fscanf(fin, "%f_%f", &xx, &yy);
    xmin=xmax=xx;
    ymin=ymax=yy;
    do {
        n=fscanf(fin, "%f_%f", &xx, &yy);
        if (yy < ymin) {
            ymin=yy;
            xmin=xx;
        }
        if (yy > ymax) {
            ymax=yy;
            xmax=xx;
        }
    } while(n==2);
    fclose(fin);
    printf("el_minimo_global_de_la_funcion_es_(%g,%g)\n",
        xmin, ymin);
    printf("el_maximo_global_de_la_funcion_es_(%g,%g)\n",
        xmax, ymax);
    return 0;
}

```