Maestría en Ciencias con Especialidad en Computación y Matemáticas Industriales
Proceso de admisión 2021
Centro de Investigación en Matemáticas (CIMAT), A.C.
Examen de programación
(Tiempo: 2.5 horas)

Nombre: ______________________________

Fecha: ______________________________

Instrucciones:

- Escriba lo más claro posible para que al digitalizar las respuestas sean legibles.
- Resuelva cada problema en hojas independientes.
- Escriba su nombre en cada hoja.
- No se debería de pasar más de 30 minutos en cada problema.
- Escriba el código lo más claramente posible y especifique el lenguaje/pseudo-código utilizado.
- Si requiere funciones estándar como determinar el máximo o mínimo de dos valores (max(a, b), min(a,b)), ordenar (sort), determinar el tamaño de una cadena de caracteres (strlen), obtener el valor absoluto (abs), raíz cuadrada (sqrt) o logaritmos (log) puede utilizarlas sin implementarlas. Será necesario que implemente cualquier función diferente a las mencionadas anteriormente.

Problema 1 [1 point]

Sea $0 < N < 50$ un entero, implementar una función que tome a N de argumento y imprima los N primeros números primos.

Ejemplo: ante la ejecución de la función con $N = 5$, se imprimirá $2, 3, 5, 7, 11$.

Problema 2 [2 point]

Desarrolle una función que reciba como parámetros dos matrices, A y B, de dimensiones $n \times n$, y devuelva si la matriz A es la inversa de B.

Ejemplo: Si la función recibe $A = \begin{pmatrix} 1 & -2 \\ -3 & 5 \end{pmatrix}$ y $B = \begin{pmatrix} -5 & -2 \\ -3 & -1 \end{pmatrix}$ la función debe devolver True, mientras que si recibe $A = \begin{pmatrix} 1 & -2 \\ -3 & 5 \end{pmatrix}$ y $B = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$ debe devolver False.

Problema 3 [2 point]

Dado un conjunto de $n+1$ puntos (x_i, y_i) para $i = 0, 1, ..., n$, tales que $(x_i = x_j) \implies (i = j)$, existe un único polinomio de grado inferior o igual a n que pasa por dichos puntos. Si denotamos por $p_{i,j}(x)$ al único polinomio de grado inferior o igual a $j - i$, que pasa por los puntos (x_k, y_k) para $k = i, i+1, ..., j$, se puede mostrar que se cumplen las siguientes recurrencias:

$p_{i,i}(x) = y_i, 0 \leq i \leq n$

$p_{i,j}(x) = \frac{(x-x_j)p_{i,j-1}(x)-(x-x_i)p_{i+1,j}(x)}{x_i-x_j}, 0 \leq i < j \leq n$

a) Utilice la recurrencia anterior para generar una función que reciba como parámetros los $n+1$ puntos, así como un valor t en donde se quiere evaluar el polinomio de grado $\leq n$ que pasa por los n puntos, y devuelva $p_{0,n}(t)$.

Ejemplo: si la función recibe los puntos $(0,1)$ y $(1,3)$, y $t = 2$, la función debe devolver el valor 5.

b) Utilice la recurrencia anterior para generar una función que reciba como parámetros los $n+1$ puntos, y devuelva el polinomio $p_{0,n}(x)$. Indique claramente la estructura de datos que utilizará para almacenar el polinomio.

Ejemplo: si la función recibe los puntos $(0,1)$ y $(1,3)$, la función debe devolver el polimio $p(x) = 2x+1$.

Problema 4 [1 point]

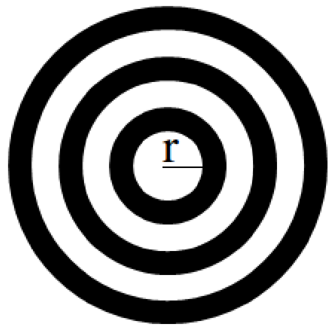
En el juego tres en línea o gato, se dispone de un tablero de 3×3, y dos jugadores se alternan para poner cruces (el primer jugador) o círculos (el segundo jugador). Los jugadores no pueden saltarse su turno, por lo que siempre deben poner su símbolo en alguna de las casillas. El juego finaliza cuando un jugador consigue poner su símbolo en toda una línea vértical, horizontal o diagonal de 3 casillas, o cuando ya no quedan casillas libres. Al inicio del juego todas las casillas están libres.

Un estado en este juego se puede representar con una matriz de 3×3, en la que las posiciones de la matriz pueden tomar los valores -1, 0 ó 1, para representar cruces, vacío o círculos, respectivamente. Desarrolle una función que reciba como parámetro el estado del tablero y devuelva un entero en función de las características del tablero:

- 1: si es un estado al que no se puede llegar siguiendo las reglas. Ejemplo: $\begin{vmatrix} X & X & X \\ & & \\ & & \end{vmatrix}$
- 2: si es un estado al que se puede llegar, y gana el primer jugador. Ejemplo: $\begin{vmatrix} X & O & O \\ & X & \\ & & X \end{vmatrix}$
- 3: si es un estado al que se puede llegar, y gana el segundo jugador. Ejemplo: $\begin{vmatrix} X & X & O \\ & X & O \\ & & O \end{vmatrix}$
- 4: si es un estado al que se puede llegar, pero aún no gana ningún jugador. Ejemplo: $\begin{vmatrix} X & X & \\ & O & O \\ & & \end{vmatrix}$

Problema 5 [2 point]

Se dispone de una cierta cantidad (P) de ml de pintura negra para pintar un conjunto de anillos concéntricos sobre un fondo blanco. Se empieza pintando un anillo negro de 1 cm de ancho, que envuelve a un círculo blanco de radio r (ver figura adjunta). A continuación en cada paso, y mientras quede pintura suficiente para pintar el siguiente anillo, se deja un anillo de un cm. de ancho sin pintar, y se pinta otro anillo. Cada anillo que se pinta tiene anchura de 1 cm y se requiere 1 ml por cada π cm^2 que hay que pintar.



a) Implemente una función **int getAnillos(double P, double r)** que, teniendo en cuenta los valores de P y r, devuelva el número de anillos que se pueden pintar.

b) Implemente la función, para que devuelva la solución en tiempo constante, es decir, que el tiempo requirido para realizar el cálculo no dependa de los valores P y r.

Problema 6 [2 point]

Como se vio en un problema anterior, dado $n+1$ puntos (x_i, y_i) para $i = 0, 1, ..., n$, tales que $(x_i = x_j) \implies (i = j)$, existe un único polinomio de grado inferior o igual a n que pasa por dichos puntos. Nos enfocamos aquí al caso $n = 2$. La propiedad. para este caso, dice que dada una tripleta $(x_1, y_1), (x_2, y_2), (x_3, y_3)$, existe una única parábola $y = ax^2 + bx + c$ pasando por estos puntos.

(a) Demostrar que los coeficientes a, b, c satisfacen un sistema lineal 3×3.

(b) Implementar un algoritmo para resolver el sistema lineal mencionado en el (a).