



Primer examen parcial de Análisis y Diseño de Algoritmos

Profesor: Miguel Ángel Vargas Lomelí

Nombre del Alumno: _____

Instrucciones: Halle la función $T(n)$ y su respectivo orden de ejecución $O(f(n))$ para los siguientes dos pseudocódigos.

1. // Dados previamente los valores de n (tamaño de la matriz), $A[n][n]$ y $B[n][n]$

```
for (i=0; i<n; i++)
    for (j=0; j<n; j++){
        C[i][j] = 0;
        for (k=0; k<n; k++)
            C[i][j] = C[i][j]+A[i][k]*B[k][j];
    }

for (i=0; i<n; i++){
    for (j=0; j<n; j++)
        printf(C[i][j], "\t");
    printf("\n");
}
```

2. Retome la idea del algoritmo de búsqueda binaria visto en clase, para el método de la bisección

```
// Método de la bisección para hallar ceros de funciones
// búsqueda binaria para hallar x tal que f(x) = 0
// Dado el valor de a, b (b>a, a>0), búsqueda de f(x)=0 para algún x en [a, b].
// Suponga que el intervalo [a, b] tiene un tamaño de "n".

fa = a*a;
fb = b*b;
encontrado = false;
do{
    n = b-a;
    c = (a+b)/2; // Punto medio del intervalo [a, b] (Similar n/2)
    fc = c*c;
    if (fa*fc<0) // Caso promedio
        b = c; // Intervalo [a, b] pasa a [a, c]. Divide el intervalo en 2
    else
        if (fb*fc<0) // Peor de los casos
            a = c; // Intervalo [a, b] pasa a [c, b]. Divide el intervalo en 2
        else
            encontrado = true;
    fa = a*a;
    fb = b*b;
while (!encontrado && (n>0.00001)); // Terminación análoga búsqueda binaria, n=1

if (encontrado) // encontrado == true
    printf("x = ", c);
else
    printf("No hallé solución");
```

Subir sus respuestas en formato pdf en plataforma de Microsoft Teams donde fue asignado su examen. Se verificarán códigos entre compañeros y en foros de internet para evitar plagios, en caso de encontrar plagios o casualmente respuestas iguales se procederá a defensa de sus anotaciones en formato de examen oral en videoconferencia.

NOTA: El GENIO se hace con 1 % de TALENTO, y un 99 % de TRABAJO.

