



Facultad de Ingeniería UNAM  
Estructura de Datos y Algoritmos I  
Semestre 2024-1  
Profesor: M.C. Leonardo Ledesma Domínguez  
Examen de Laboratorio

El examen de laboratorio consistirá en 3 ejercicios de programación práctica en Python o C (elección del alumno) que tendrán como objetivo evaluar el contenido de las 12 prácticas de la asignatura de Estructura de Datos y Algoritmos I realizadas durante el semestre.

El valor de este examen tendrá un valor de 40% de calificación de laboratorio.

Dicha evaluación contendrá y evaluará todas las prácticas contenidas en el manual.

1. Nivel de dificultad: Fácil (20 puntos) – *Stacks Wars*

Considere 3 *Stacks* A, B y C llenadas de números aleatorios entre 1-100.

Genere un orden de turnos utilizando una *Circular Queue*. Una vez asignado un turno aleatorio, el *Stack* actual, puede argumentar; que el número en su *top* es mayor, menor o igual a los tops de las otras pilas.

- Si es Falso, adquiere los tops de las pilas vecinas en su lista de números.
- Si es Correcto, se elimina su elemento top.

Programa una estrategia de adivinanza para las *Stacks*:

- Un *Stack* que siempre diga que su número es mayor cada 2 turnos y en su tercer turno que su número es menor y se repite.
- Un *Stack* que escoga de manera aleatoria su adivinanza.
- Un *Stack* que varíe sus adivinanzas progresivamente, eg. primer turno mayor, segundo turno menor y tercer turno igual y se repite

Salida: El programa deberá mostrar cada iteración y el *Stack* ganador.

2. Nivel de dificultad: Media (30 puntos) – *Policemen & Thieves*

Dada una *Lista Circular Doblemente Ligada L* cuyos elementos son P (Policeman) o T (Thief) de tamaño  $n$ . Considere que un policía puede atrapar a solo un ladrón y que cada policía no puede atrapar a un ladrón que está a  $K$  unidades de distancia de él.

Llene de manera aleatoria dicha lista de tamaño  $n$ . Puede considerar una  $n \geq 5$  y  $n \leq 15$ .

***Se necesita encontrar el máximo número de ladrones a ser capturados e imprima las parejas formadas en ese caso.***

**Utilice un algoritmo de fuerza bruta o greedy solamente.**

Ejemplo: Sea  $CDLL = \{'T', 'T', 'P', 'P', 'P', 'P', 'T', 'T'\}$  y  $K = 2$ .

Máximo Número de Ladrones capturados será = 4

$CDLL[2] - CDLL[0]$

$CDLL[3] - CDLL[1]$

$CDLL[4] - CDLL[6]$

$CDLL[5] - CDLL[7]$

Para la misma  $CDLL$  pero con  $K=1$ :

Máximo Número de Ladrones capturados será = 2

$CDLL[2] - CDLL[1]$

$CDLL[5] - CDLL[6]$

Programa de manera recursiva la siguiente figura en consola (n es una potencia de 2, incluido el 1):

Output :

Output :

Output :

[illegible]