

TALLER EVALUADO (30%)

EJERCICIO 01 (10%):

Dada una matriz cuadrada $N \times N$, con N mayor a cero, cuyos elementos son todos enteros, se desea calcular el máximo de las dos diagonales principales. Por ejemplo, si tenemos la siguiente matriz, donde las diagonales principales, están marcadas en negrita.

$$\begin{vmatrix} \mathbf{5} & 5 & -3 & \mathbf{0} \\ 11 & \mathbf{8} & \mathbf{8} & 3 \\ 0 & -2 & \mathbf{14} & 4 \\ \mathbf{1} & -7 & 0 & \mathbf{10} \end{vmatrix}$$

El máximo reportado es 14.

Para responder a esta pregunta, realice lo siguiente:

1. Descargue el esqueleto de algoritmo identificado como “examenP1.gcl” y complete las acciones que allí son indicadas. Guarde su algoritmo como “examenP1_XXXXX.gcl” donde XXXXX son los dígitos de su carnet.
2. Traduzca el algoritmo a un programa en python equivalente. Usted debe escribir el invariante y la función de cota de cada ciclo utilizado en la sección de cálculo, también las especificaciones de la pre y postcondiciones. No debe escribir el invariante para los ciclos de lectura o escritura, pero si la función de cota. Incluya las precondiciones y postcondiciones. Además, debe hacer una programación por contrato. Guarde su programa como “examenP2_XXXXX.py”
3. Suba el algoritmo y programa en el aula virtual.

EJERCICIO 02 (10%):

Se dice que hay un ascenso, en una secuencia S de enteros, cuando la posición $S[i-1] < S[i]$. Suponga que se tiene una secuencia de enteros positivos terminada en 0, que no es ni vacía ni unitaria. Se desea calcular el número de ascensos de la secuencia. Por ejemplo en la secuencia $S_1=(7,4,3,3,8,5,9,10)$ hay 3 ascensos, mientras que en $S_2=(17,15,12,8,8,5,1,1)$ hay 0 ascensos.

Para responder a esta pregunta, realice lo siguiente:

1. Descargue el esqueleto de algoritmo identificado como “examenP2.gcl” y complete las acciones que allí son indicadas. Guarde su algoritmo como “examenP2_XXXXX.gcl” donde XXXXX son los dígitos de su carnet.
2. Traduzca el algoritmo a un programa en python equivalente. No use arreglos ni ciclos for. No utilice try-except para la verificación de la pre y postcondiciones, ni de los invariantes y cotas. Hágalo solamente con la instrucción assert. No debe escribir el invariante para los ciclos de lectura o escritura, pero sí la función de cota. Guarde su programa como “examenP2_XXXXX.py”
3. Suba el algoritmo y programa en el aula virtual.

EJERCICIO 03 (10%):

En los juegos olímpicos de invierno que acaban de finalizar participaron 92 países en 7 disciplinas deportivas, con un total de 2920 atletas y 102 eventos deportivos (Wikipedia, 2018). Se desea hacer un programa en PYTHON para obtener algunas estadísticas referentes a estos juegos.

Para ello se le pide que proponga una estructura denominada Pais que almacene los siguientes datos: nombre del país (nombre), número de atletas participantes (numatl), número de medallas de oro obtenidas (numoro), número de medallas de plata obtenidas (numpla), número de medallas de bronce obtenidas (numbro). Coloque a los campos de la estructura los identificadores que aparecen entre parentesis

El programa debe escribir en pantalla el país con el número máximo de medallas obtenidas, indicando el número obtenido por cada categoría (oro, plata y bronce). Además, el total de medallas otorgadas en cada categoría. Siga el siguiente formato de salida:

```
PAIS CON MAXIMO NUMERO DE MEDALLAS: NNNNNNN
    ORO = XX  PLATA = XX  BRONCE = XX
NUMERO TOTAL DE MEDALLAS OTORGADAS:
    ORO = XX  PLATA = XX  BRONCE = XX
```

Para probar su programa coloque un total de países (MAX) igual a 5, declarando a MAX como una constante. Su programa debe ser lo suficientemente general para que si se quieren hacer pruebas con un mayor número de países, sólo sea necesario cambiar ese valor.

Utilice programación robusta en la lectura de datos. No debe escribir ningún invariante en los ciclos pero si la función de cota. Escriba las pre y postcondiciones del programa.

Guarde su programa como “examenP3_XXXXX.py” donde XXXXX son los dígitos de su carnet y súbalo en el aula virtual.