

Indicaciones específicas:

- Esta evaluación contiene 8 páginas (incluyendo esta página) con 3 preguntas. El total de puntos son 20.
- El tiempo límite para la evaluación es 100 minutos.
- Cada pregunta deberá ser respondida en un solo archivo con el número de la pregunta y tu código de estudiante. Por ejemplo:
 1. p1_2020010202.py
 2. p2_2020010202.py
 3. p3_2020010202.py
- Luego deberás incluir estos archivos en una carpeta con nombre **pc3**; para que finalmente envíes esta carpeta comprimida **pc3.zip** a www.gradescope.com

Competencias:

- Para los alumnos de la carrera de Ciencia de la Computación
 - Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina. (**Usar**)
- Para los alumnos de las carreras de Ingeniería
 - Capacidad de aplicar conocimientos de ingeniería (**nivel 2**).

Calificación:

Tabla de puntos (sólo para uso del professor)

Question	Points	Score
1	6	
2	7	
3	7	
Total:	20	

1. (6 points) Diseñe e implemente un programa que permita crear un diccionario con los siguientes datos:

```
reniec = {  
    "10345678" : "Jaime",  
    "23456789" : "Pedro",  
    "46732212" : "Carlos",  
    "76554434" : "Macarena",  
    "34567890" : "Julia",  
    "67554875" : "Pedro",  
    "76543321" : "Julia",  
    "76765432" : "Carlos",  
    "87654323" : "Pedro"  
}
```

y luego el programa realice lo siguiente:

- Mostrar los datos ordenados de menor a mayor según el dni
- Buscar un dato según el dni, el mismo que se ingresa como dato desde el teclado. Si el dni no se encuentra en el diccionario dar un mensaje que indique "El DNI ingresado no esta registrado en el diccionario"
- Pedir un nombre como dato, y formar una nueva lista con todos los dni de las personas que tienen ese nombre.

Algunos ejemplos de diálogo de este programa serían:

Listing 1: Ejemplo 1

```
Ordenados por el DNI  
  
10345678    Jaime  
23456789    Pedro  
34567890    Julia  
46732212    Carlos  
67554875    Pedro  
76543321    Julia  
76554434    Macarena  
76765432    Carlos  
87654323    Pedro  
  
DNI a buscar: 76543321  
El nombre asociado al DNI es de  Julia  
  
Nombre : Pedro  
['23456789', '67554875', '87654323']
```

Listing 2: Ejemplo 1

Ordenados por el DNI

10345678	Jaime
23456789	Pedro
34567890	Julia
46732212	Carlos
67554875	Pedro
76543321	Julia
76554434	Macarena
76765432	Carlos
87654323	Pedro

DNI a buscar: 7777777

El DNI ingresado no esta registrado en el diccionario

Nombre : Carlos

['46732212', '76765432']

Listing 3: Ejemplo 1

Ordenados por el DNI

10345678	Jaime
23456789	Pedro
34567890	Julia
46732212	Carlos
67554875	Pedro
76543321	Julia
76554434	Macarena
76765432	Carlos
87654323	Pedro

DNI a buscar: 67554875

El nombre asociado al DNI es de Pedro

Nombre : Sergio

[]

La rúbrica para esta pregunta es:

Criterio	Logrado	Parcialmente Logrado	No Logrado
Algoritmo	Es preciso, finito y hace exactamente lo que el enunciado requiere (4pts)	Es preciso, finito y hace la mitad o más de lo que el enunciado requiere (1pts)	Hace menos de la mitad de lo que el enunciado requiere (0pts)
Sintaxis	Todas las sentencias son correctas y no hay errores de sintaxis (1pts)	Más de la mitad de las sentencias son correctas y no hay errores de sintaxis (0.5pts)	Menos de la mitad de las sentencias son correctas (0pts)
Legible	El algoritmo es correcto y el nombre de todas las variables y funciones son descriptivas (1pts)	El algoritmo es correcto y el nombre de la mitad de las variables y funciones son descriptivas (0.5 pts)	El algoritmo es correcto y el nombre de menos la mitad de las variables y funciones son descriptivas (0 pts)

2. (7 points) Realice un programa que a través de una **función recursiva** permita hallar la **suma de los cuadrados** de los **n** primeros números naturales.

Si $n=5$ la suma sería

$$1^2 + 2^2 + 3^2 + 4^2 + 5^2 = 55$$

Si $n=7$ la suma sería

$$1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + 7^2 = 140$$

Para asignar el puntaje a esta pregunta, es absolutamente necesario que realice la función recursiva.

- Luego: imagine que calcula lo mismo, pero esta vez, el algoritmo es "NO" recursivo. Analice la **complejidad algorítmica**, en este nuevo escenario e indique el **"tipo"** según la complejidad algorítmica Big(O).

Algunos ejemplos de diálogo de este programa serían:

Listing 4: Ejemplo 1

```

Input :
n : 7
Output:
La sumatoria es 140

Si el algoritmo NO fuese recursivo, la complejidad algorítmica
sería del tipo .....

```

Listing 5: Ejemplo 1

```

Input :
n : 12
Output:
La sumatoria es 650

Si el algortimo NO fuese recursivo, la complejidad algoritmica
seria del tipo .....

```

La rúbrica para esta pregunta es:

Criterio	Logrado	Parcialmente Logrado	No Logrado
Algo-ritmo	Es preciso, finito y hace exactamente lo que el enunciado requiere (3pts)	Es preciso, finito y hace la mitad o más de lo que el enunciado requiere (1.5pts)	Hace menos de la mitad de lo que el enunciado requiere (0pts)
Sintáxis	Todas las sentencias son correctas y no hay errores de sintáxis (1pts)	Más de la mitad de las sentencias son correctas y no hay errores de sintáxis (0.5pts)	Menos de la mitad de las sentencias son correctas (0pts)
Legible	El algoritmo es correcto y el nombre de todas las variables y funciones son descriptivas (1pts)	El algoritmo es correcto y el nombre de la mitad de las variables y funciones son descriptivas (0.5 pts)	El algoritmo es correcto y el nombre de menos la mitad de las variables y funciones son descriptivas (0 pts)
Funciones	Se implementan y usan correctamente todas las funciones requeridas (1pts)		Las funciones no están implementadas o usadas correctamente (0pts)
Complejidad Algoritmica	Es correcta la complejidad algorítmica señalada (1pts)		No describe o es incorrecta la complejidad algorítmica señalada. (0pts)

3. (7 points) Enunciado Diseñe e implemente un programa que permita crear una matriz de orden: $\mathbf{n} \times \mathbf{m}$, con datos aleatorios. Los valores de \mathbf{n} y \mathbf{m} se leen desde el teclado y son mayores a uno. El programa debe:
- Imprimir la matriz y luego
 - Hallar la suma de los elementos ubicados en el borde de la matriz.
 - Finalmente, analice la **complejidad algorítmica**, del código que en su programa permite *imprimir la matriz* e indique el **"tipo"** según la complejidad algorítmica Big(O).

Algunos ejemplos de diálogo de este programa serían:

Listing 6: Ejemplo 1

```
Input:

Filas : 5
Columnas : 6

Output:
  2   9   7   2   6   1
  7   8   4   7   8   5
  4   7   8   9   3   6
  2   6   9   3   4   2
  6   3   9   8   3   8
La suma de los elementos del borde de la matriz es:    90
-----
La complejidad algoritmica del codigo que permite imprimir la
matriz es del tipo .....
```

Listing 7: Ejemplo 1

```
Input:

Filas : 7
Columnas : 5

Output:
  4   1   3   5   4
  1   5   3   9   4
  2   9   8   6   9
  4   2   3   3   6
  3   6   3   7   5
  6   8   6   9   9
  4   6   3   7   3
La suma de los elementos del borde de la matriz es:    89
-----
La complejidad algoritmica del codigo que permite imprimir la
matriz es del tipo .....
```

La rúbrica para esta pregunta es:

Criterio	Logrado	Parcialmente Logrado	No Logrado
Algoritmo	Es preciso, finito y hace exactamente lo que el enunciado requiere (3pts)	Es preciso, finito y hace la mitad o más de lo que el enunciado requiere (1pts)	Hace menos de la mitad de lo que el enunciado requiere (0pts)
Sintaxis	Todas las sentencias son correctas y no hay errores de sintaxis (1pts)	Mas de la mitad de las sentencias son correctas y no hay errores de sintaxis (0.5pts)	Menos de la mitad de las sentencias son correctas (0pts)
Legible	El algoritmo es correcto y el nombre de todas las variables y funciones son descriptivas (1pts)	El algoritmo es correcto y el nombre de la mitad de las variables y funciones son descriptivas (0.5 pts)	El algoritmo es correcto y el nombre de menos la mitad de las variables y funciones son descriptivas (0 pts)
Iteración	Recorre adecuadamente los elementos de la matriz (1pts)		No recorre los elementos de la matriz de forma programática y solo funciona en algunos casos (0 pts).
Complejidad Algoritmica	Es correcta la complejidad algorítmica señalada (1pts)		No describe o es incorrecta la complejidad algorítmica señalada. (0pts)