

Indicaciones específicas:

- Esta evaluación contiene 7 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) Implementar un algoritmo que permita procesar la tabla del número de días feriados por país.

Colombia	20
Venezuela	14
Ecuador	13
Peru	13
Chile	20
Brasil	14
Argentina	16
Paraguay	12
Uruguay	16
Bolivia	11

1. Crear un diccionario con la información de la tabla. Los siguientes items deben procesar el diccionario para obtener los resultados.
2. El usuario ingresa un nombre de país y el programa imprime la cantidad de días feriados si este se encuentra en el diccionario. Caso contrario imprime el mensaje "El nombre del país no se encuentra en nuestros datos".
3. El usuario ingresa un número de días N y el programa imprime todos los países que tienen más de N días feriados. Indicar la complejidad algorítmica.
4. Crear un nuevo diccionario con los países que tienen más de N días feriados. Usando el nuevo diccionario calcular e imprimir la suma total de feriados.

Algunos ejemplos de diálogo de este programa serían:

Listing 1: Ejemplo 1

```
Input:
Por favor ingrese un nombre de país: Peru
Por favor ingrese un número de días: 19
Output:
Los feriados de Peru son 13
Los países que tienen mas de 19 feriados son: Colombia Chile
El total de feriados es: 40
```

Listing 2: Ejemplo 2

```
Input:
Por favor ingrese un nombre de país: perues
Por favor ingrese un número de días: 100
Output:
El nombre del país no se encuentra en nuestros datos
Los países que tienen mas de 100 feriados son:
El total de feriados es: 0
```

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) Implementar una función recursiva que imprima cada dígito de un número ingresado por el usuario.

- Este número debe ser entero y mayor a 10.
- Calcule la suma de los dígitos y la imprima.
- Indicar la complejidad algorítmica.

Algunos ejemplos de diálogo de este programa serían:

Listing 3: Ejemplo 1

```

Ingrese numero: 12345
Digito 0, 5
Digito 1, 4
Digito 2, 3
Digito 3, 2
Digito 4, 1
Suma: 15

```

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 (1.5pts)	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)
Funciones	Se implementan y usan correctamente todas las funciones requeridas (1pts)		Las funciones no están implementadas o usadas correctamente (0pts)
Complejidad Algorítmica	Es correcta la complejidad algorítmica señalada (1pts)		No describe o es incorrecta la complejidad algorítmica señalada. (0pts)

3. (7 points) Escribir un programa que cree una matriz con las siguientes características:

- El usuario ingresa un valor N mayor a 3, y una variable sector que puede tener el valor de "superior" o "inferior".
- Crear una matriz $N \times N$ con valores enteros aleatorios (del 0 al 9).
- Imprimir la matriz
- Calcular la suma de todos los elementos de la matriz. Imprimir la suma.
- Implementar una función que reciba la matriz y dependiendo de la variable sector, si fuera "superior" los elementos superiores a la diagonal serán cambiados por "@" y los inferiores por "*". Si fuera "inferior" será al contrario. Reemplaza la diagonal por el valor de la suma dividido entre 100.

Esta función debe recibir la suma como parámetro.

Indicar la complejidad algorítmica de esta función.

- Imprimir la matriz final.

Algunos ejemplos de diálogo de este programa serían:

Listing 4: Ejemplo 1

```

Por favor ingrese N: 4
Por favor ingrese sector (superior o inferior): superior

```

```
Matriz original (aleatoria):
2 3 6 1
4 5 2 9
8 7 3 0
3 9 1 2
Suma: 65
Matriz final:
0.65 @ @ @
* 0.65 @ @
* * 0.65 @
* * * 0.65
```

Listing 5: Ejemplo 1

```
Por favor ingrese N: 4
Por favor ingrese sector (superior o inferior): inferior
Matriz original (aleatoria):
2 3 6 1
4 5 2 9
8 7 3 0
3 9 1 2
Suma: 65
Matriz final:
0.65 * * *
@ 0.65 * *
@ @ 0.65 *
@ @ @ 0.65
```

Listing 6: Ejemplo 1

```
Input:
Output:
```

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)