

## Evaluación Sumativa 02: Introducción a la Programación Segura (35%)

<b>ÁREA ACADÉMICA</b>	Informática y Telecomunicaciones	<b>CARRERA</b>	Ingeniería en Informática
<b>ASIGNATURA</b>	Introducción a la Programación Segura	<b>CÓDIGO</b>	TI3V11
<b>SEDE</b>	SANTIAGO CENTRO	<b>DOCENTE</b>	Luis Yáñez Carreño
<b>Unidad de Aprendizaje</b>	Nº2 - 3	<b>Criterios a Evaluar</b>	2.1.1 - 2.2.2 - 2.1.3- 2.2.4 - 3.1.1 - 3.1.2 - 3.1.3 - 3.1.4 - 3.1.5
<b>DURACIÓN</b>	120 minutos	<b>FECHA</b>	

<b>NOMBRE ESTUDIANTE:</b>			
Apellido Paterno		Apellido Materno	
Nombres			
<b>RUT:</b>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<b>PUNTAJE MÁXIMO</b>	108	<b>NOTA:</b>	
<b>PUNTAJE OBTENIDO</b>			Firma conforme
<b>Solicita re-corrección</b>	<b>Sí</b>	<b>No</b>	<b>Motivo:</b>

### INSTRUCCIONES GENERALES:

- La nota 4.0 se obtiene logrando un 60% del puntaje total.
- Algoritmo debe ser realizado en lenguaje de programación Python.
- Se prohíbe el uso de celulares, mp3, mp4, iphone, ipod o similares durante la evaluación.
- Podrá consultar los PDF utilizados en clase.
- El nombre del algoritmo principal y el de funciones debe ser según la siguiente Estructura: EVA2\_ApellidoNombre\_Vespertino, EVA2Func\_ApellidoNombre .
- Debe comprimir los archivos identificando el comprimido con el mismo nombre del algoritmo principal (EVA2\_ApellidoNombre\_Vespertino).
- El archivo debe ser subido a un recurso de tarea creado para tal efecto en moodle.
- Como requisito de evaluación y con la finalidad de desarrollar la lógica de la programación, **NO puede utilizar BREAK ni CONTINUE. Por cada utilización de alguno de estos será descontado 1 punto de la nota obtenida.**

## DESCRIPCIÓN DE LA ACTIVIDAD:

Los Estudiantes, individualmente, deben modificar el siguiente algoritmo:

```
cont=0
suma=0
nota=-1
rut=input("ingrese rut del Estudiante:")
while nota!=0:
    nota=int(input("Ingrese nota del Estudiante:"))
    while (nota<10 or nota>70) and nota!=0:
        nota=int(input("ERROR - Ingrese un correcta"))
    suma=suma+nota
    cont=cont+1
promedio=suma/(cont-1)
print("El promedio es:",promedio)
if promedio>50:
    print("El estudiante Aprueba la Asignatura")
elif promedio<40:
    print("El estudiante Reprueba la Asignatura")
else:
    print("El estudiante debe rendir Examen")
```

Cuya Ejecución permite Ingresar y calcular el promedio de notas de un estudiante. Para lo cual debe Digitar el rut y notas correspondientes, el ingreso de estas debe ser realizado hasta que se ingrese un 0 (cero) como indicador de fin de ingreso.

Una vez finalizado el registro de las notas, se obtendrá como información en pantalla el promedio obtenido y un mensaje que indica si el estudiante REPRUEBA, APRUEBA o debe RENDIR EXAMEN, para lo cual debe considerar:

- Promedio de notas mayor a 5.0, el mensaje entregado debe ser APRUEBA ASIGNATURA.
- Promedio de notas menor a 4.0, el mensaje entregado debe ser REPRUEBA ASIGNATURA.
- Promedio de notas mayores o igual a 4.0 y menor o igual a 5.0, el mensaje entregado debe ser DEBE RENDIR EXAMEN.

Las modificaciones deberán permitir cumplir con los siguientes requerimientos:

El algoritmo debe permitir lo siguiente:

- Al iniciar la ejecución mostrar el siguiente menú de opciones:

**1. Iniciar Registro      2. Consultar Estudiante      3. Eliminar Estudiante      4. Salir**

El usuario deberá seleccionar el número que corresponde a la acción que desea realizar y donde la única forma de finalizar la ejecución es con la opción 4

- Al Seleccionar la opción 1 (Iniciar Registro) solicitará el rut del Estudiante:

**Ingreso Rut del Estudiante :** 88888888-8

Para seguir avanzando debe realizar las siguientes validaciones:

- De las validaciones contenidas en este punto, debe seleccionar una comentando en el código la seleccionada:
    - El rut ingresado no debe existir registrado en la TUPLA llamada TuplaApellidoNombre, la cual debe ser creada con los siguientes datos: 11111111-1, 22222222-2, 33333333-3, 44444444-4, 55555555-5 y 66666666-6
    - El rut debe ser válido según algoritmo de validación creado en clase que lee el string del ingreso de rut como una lista de caracteres.  
La validación seleccionada debe encontrarse en una función dentro del archivo correspondiente
  - De las validaciones contenidas en este punto, debe seleccionar una comentando en el código la seleccionada:
    - El rut debe ser ingresado con los 10 caracteres (12345678-9), si un rut es menor a 10 millones automáticamente se agrega un 0 al inicio.
    - El rut solo debe permitir valores numéricos, el guión y la K.
    - El rut solo debe permitir el ingreso del guión en la posición 8 del string y el Dígito en la 9
  - Para cada error de ingreso de rut debe solicitar reingreso al usuario hasta que cumpla con los requisitos para continuar la venta.
- Una vez validado el rut el usuario debe ingresar la asignatura y las notas hasta que sea ingresada una con valor 0 (cero). Luego de digitar cada nota debe presionar la tecla ENTER para poder ingresar el siguiente valor:
  -

**Ingreso Asignatura:** Fundamentos

**Ingreso nota:** 1.0

**Ingreso nota:** 3.5

**Ingreso nota:** 5.5

**Ingreso nota:** 0

- Validación de Notas: Las notas ingresadas deben encontrarse entre el rango de 1.0 y 7.0, cualquier otra debe ser considerada error, pedir su reingreso y no considerarla para el promedio:

Ingrese nota: 1.0

Ingrese nota: 8.0

Error Ingrese nota nuevamente: 7.5

Error Ingrese nota nuevamente: 3.5

Ingrese nota: 5.5

Ingrese nota: 0

- Consultar si desea agregar otro estudiante:

Desea agregar otra Asignatura (SI/NO) SI

- Cada asignatura debe ser agregada a una lista llamada ASIG\_ApellidoNombre
- Debe calcular el promedio de las notas de cada asignatura
- Si la respuesta es SI debe reiniciar desde la solicitud de asignatura y notas

Ingrese Asignatura: Bases de Datos

Ingrese nota: 1.0

Ingrese nota: 3.5

Ingrese nota: 5.5

Ingrese nota: 0

Desea agregar otra Asignatura (SI/NO) NO

- Si la respuesta es NO deberá Volver a Reiniciar desde la aparición del Menú, permitiendo seleccionar cualquier opción.

**1. Iniciar Registro**

**2. Consultar Estudiante**

**3. Eliminar Estudiante**

**4. Salir**

- Al finalizar el registro de cada asignatura, se agregará a un diccionario llamado DIC\_ApellidoNombre la lista de asignaturas (ASIG\_ApellidoNombre) registrada, relacionándola al rut del estudiante y a otro diccionario llamado Total\_ApellidoNombre el promedio general de todas las asignaturas ingresadas para el Estudiante relacionándola al rut.
- Al seleccionar la opción 2 (Consultar Estudiante) El usuario debe ingresar el rut del Estudiante a consultar:

Ingrese rut a consultar: 88888888-8

- El software debe buscar el rut consultado en ambos diccionarios para mostrar la información de las asignaturas y el promedio correspondiente.

**El Estudiante rut 88888888-8 registra las siguientes asignaturas: FUNDAMENTOS, BASE DE DATOS, el promedio general de todas las asignaturas es 47**

- En caso de no existir el rut en el diccionario deberá indicarlo con un mensaje.

**El Estudiante consultado no se encuentra registrado**

- Luego de mostrar la información debe volver al menú:

**1. Iniciar Registro      2. Consultar Estudiante      3. Eliminar Estudiante      4. Salir**

- Al seleccionar la opción 3 (Eliminar Estudiante) El usuario debe ingresar el rut del a eliminar:

**Ingrese rut a Eliminar: 88888888-8**

- El software debe buscar el rut consultado en ambos diccionarios y si existe eliminarlo enviando un mensaje:

**El Estudiante rut 88888888-8 ha sido eliminado**

- En caso de no existir el rut en el diccionario deberá indicarlo con un mensaje.

**El Estudiante a eliminar no se encuentra registrado**

- Luego de mostrar la información debe volver al menú:

**1. Iniciar Registro      2. Consultar Estudiante      3. Eliminar Estudiante      4. Salir**

- Al seleccionar la opción 4 (Salir), la ejecución del programa finaliza enviando el siguiente mensaje:

**Sesión Finalizada**

RUBRICA

Criterio	Excelente (4)	Bueno (3)	Regular (2)	Deficiente (1)	No Realizado (0)
<b>1. Archivos que contienen algoritmo principal y de funciones son nombrados según requerimientos</b>	Todos los archivos están nombrados exactamente según los requerimientos.	La mayoría de los archivos están nombrados según los requerimientos con pocas excepciones.	Algunos archivos están nombrados correctamente, pero otros no.	Pocos archivos están nombrados correctamente según los requerimientos.	Ningún archivo está nombrado según los requerimientos.
<b>2. Algoritmo recibido se encuentra ordenado y correctamente indentado</b>	El algoritmo está perfectamente ordenado y correctamente indentado.	El algoritmo está ordenado y bien indentado con pequeños errores.	El algoritmo tiene varios errores de orden y/o indentación.	El algoritmo está desordenado y mal indentado en su mayoría.	El algoritmo está completamente desordenado y sin indentación.
<b>3. Inicializa todas las variables que utiliza y nombra estructuras según requerimientos</b>	Todas las variables están correctamente inicializadas y nombradas según los requerimientos.	La mayoría de las variables están correctamente inicializadas y nombradas con pocas excepciones.	Algunas variables están correctamente inicializadas y nombradas, pero otras no.	Pocas variables están correctamente inicializadas y nombradas.	Ninguna variable está correctamente inicializada o nombrada según los requerimientos.
<b>4. Recoge valores ingresados por el usuario y los almacena en variables con nombres representativos a su contenido</b>	Recoge y almacena los valores en variables con nombres representativos de manera clara y correcta.	Recoge y almacena los valores con nombres representativos con pocos errores.	Recoge y almacena los valores pero con varios errores en los nombres de las variables.	Recoge y almacena los valores pero con muchos errores en los nombres de las variables.	No recoge o almacena los valores de manera adecuada.
<b>5. Al iniciar ejecución muestra menú</b>	Muestra el menú claramente y de manera intuitiva al iniciar.	Muestra el menú de manera adecuada pero con algunos detalles de presentación.	Muestra el menú pero con errores de presentación o falta de claridad.	Muestra el menú pero es confuso o incompleto.	No muestra el menú.
<b>6. Todas las opciones de menú realizan acciones solicitadas, volviendo después de cada ejecución</b>	Todas las opciones funcionan correctamente y el programa regresa al menú después de cada acción.	La mayoría de las opciones funcionan correctamente, con pocas excepciones.	Algunas opciones funcionan, pero no todas, o no regresa al menú correctamente.	Pocas opciones funcionan correctamente y no regresa al menú.	Ninguna opción funciona correctamente o no regresa al menú.
<b>7. Algoritmo permite ingresar rut, asignatura y nota</b>	Permite el ingreso de rut, asignatura y nota de manera clara y sin errores.	Permite el ingreso de rut, asignatura y nota con pocos errores.	Permite el ingreso de rut, asignatura y nota pero con varios errores.	Permite el ingreso de solo algunos datos o con muchos errores.	No permite el ingreso de estos datos.

<b>8. Realiza y comenta correctamente validación seleccionada del punto A de validaciones</b>	Realiza y comenta la validación de manera completa y correcta.	Realiza la validación correctamente pero con comentarios incompletos.	Realiza la validación pero con errores y/o comentarios poco claros.	Realiza la validación pero no la comenta.	No realiza la validación.
<b>9. Realiza y comenta correctamente validación seleccionada del punto B de validaciones</b>	Realiza y comenta la validación de manera completa y correcta.	Realiza la validación correctamente pero con comentarios incompletos.	Realiza la validación pero con errores y/o comentarios poco claros.	Realiza la validación pero no la comenta.	No realiza la validación.
<b>10. Crea función correctamente para validación de punto A</b>	La función es creada correctamente, con código limpio y eficiente.	La función es creada correctamente pero con código mejorable.	La función es creada pero contiene algunos errores o código ineficiente.	La función es creada pero contiene muchos errores o es ineficiente.	No crea la función.
<b>11. Función retorna valor hacia algoritmo principal</b>	La función retorna el valor correctamente hacia el algoritmo principal.	La función retorna el valor con pocos errores.	La función retorna el valor pero con varios errores.	La función retorna el valor con muchos errores o de manera ineficiente.	La función no retorna ningún valor.
<b>12. Identifica errores de validación de nota y solicita reingreso</b>	Identifica errores correctamente y solicita el reingreso de manera clara y efectiva.	Identifica errores y solicita reingreso con pocos errores.	Identifica errores pero solicita reingreso con varios errores.	Identifica errores pero no solicita reingreso de manera adecuada.	No identifica errores o no solicita reingreso.
<b>13. Identifica errores de validación y solicita reingreso de rut hasta que se ingrese uno correctamente según validaciones seleccionadas</b>	Identifica errores correctamente y solicita el reingreso de manera clara y efectiva.	Identifica errores y solicita reingreso con pocos errores.	Identifica errores pero solicita reingreso con varios errores.	Identifica errores pero no solicita reingreso de manera adecuada.	No identifica errores o no solicita reingreso.
<b>14. Permite ingreso de asignaturas mientras respuesta de ingreso de nueva asignatura sea SI</b>	Permite el ingreso de nuevas asignaturas correctamente mientras la respuesta sea SI.	Permite el ingreso de nuevas asignaturas con pocos errores.	Permite el ingreso de nuevas asignaturas pero con varios errores.	Permite el ingreso de nuevas asignaturas pero con muchos errores.	No permite el ingreso de nuevas asignaturas.
<b>15. Acumula valores de notas ingresados y calcula promedio general total en variable acumulativa correctamente</b>	Acumula y calcula el promedio de manera correcta y precisa en variable acumulativa	Acumula y calcula el promedio con pocos errores en variable acumulativa	Acumula y calcula el promedio pero con varios errores.	Acumula y calcula el promedio pero con muchos errores.	No acumula o no calcula el promedio.

<b>16. Ingresar nombre de todas las asignaturas a la lista correctamente</b>	Ingresar todos los nombres de las asignaturas correctamente a la lista.	Ingresar la mayoría de los nombres correctamente.	Ingresar algunos nombres correctamente pero con varios errores.	Ingresar pocos nombres correctamente o con muchos errores.	No ingresar los nombres de las asignaturas a la lista.
<b>17. Al finalizar venta almacena en diccionario DIC_ApellidoNombre el rut del estudiante como clave y la lista de asignaturas como valor</b>	Almacena correctamente en el diccionario con la estructura solicitada.	Almacena correctamente pero con pequeños errores en la estructura.	Almacena en el diccionario pero con varios errores.	Almacena en el diccionario pero con muchos errores.	No almacena en el diccionario.
<b>18. Al finalizar registro almacena en diccionario Total_ApellidoNombre el rut del estudiante como clave y el promedio general como valor</b>	Almacena correctamente el promedio en el diccionario con la estructura solicitada.	Almacena el promedio correctamente pero con pequeños errores en la estructura.	Almacena el promedio en el diccionario pero con varios errores.	Almacena el promedio en el diccionario pero con muchos errores.	No almacena el promedio en el diccionario.
<b>19. Agrega a los diccionarios la información de todos los estudiantes y sus promedios registrados durante la ejecución del sistema</b>	Agrega correctamente toda la información de todos los estudiantes y sus promedios.	Agrega la mayoría de la información correctamente.	Agrega alguna información pero con varios errores.	Agrega poca información o con muchos errores.	No agrega la información.
<b>20. Permite ingreso de rut para búsqueda de estudiante</b>	Permite el ingreso de rut para búsqueda de manera clara y sin errores.	Permite el ingreso de rut con pocos errores.	Permite el ingreso de rut pero con varios errores.	Permite el ingreso de rut pero con muchos errores.	No permite el ingreso de rut para búsqueda.
<b>21. Busca en diccionario rut de estudiante correctamente</b>	Busca y encuentra correctamente el rut en el diccionario.	Busca y encuentra el rut con pocos errores.	Busca el rut pero con varios errores.	Busca el rut pero con muchos errores.	No busca el rut en el diccionario.
<b>22. Muestra información de estudiante encontrado según formato solicitado en ejemplo</b>	Muestra la información en el formato correcto y de manera clara.	Muestra la información correctamente pero con pequeños errores en el formato.	Muestra la información pero con varios errores en el formato.	Muestra la información pero con muchos errores en el formato.	No muestra la información.



<b>23. Si estudiante no existe en el diccionario muestra mensaje en formato solicitado en ejemplo</b>	Muestra el mensaje correcto cuando el estudiante no existe en el diccionario.	Muestra el mensaje pero con pequeños errores en el formato.	Muestra el mensaje pero con varios errores en el formato.	Muestra el mensaje pero con muchos errores en el formato.	No muestra ningún mensaje.
<b>24. Permite ingreso de rut para eliminación de estudiante</b>	Permite el ingreso de rut para eliminación de manera clara y sin errores.	Permite el ingreso de rut con pocos errores.	Permite el ingreso de rut pero con varios errores.	Permite el ingreso de rut pero con muchos errores.	No permite el ingreso de rut para eliminación.
<b>25. Elimina estudiante si existe en el diccionario</b>	Elimina correctamente al estudiante si existe en el diccionario.	Elimina al estudiante con pocos errores.	Elimina al estudiante pero con varios errores.	Elimina al estudiante pero con muchos errores.	No elimina al estudiante.
<b>26. Si estudiante no existe en diccionario muestra mensaje en formato solicitado en ejemplo</b>	Muestra el mensaje correcto cuando el estudiante no existe en el diccionario.	Muestra el mensaje pero con pequeños errores en el formato.	Muestra el mensaje pero con varios errores en el formato.	Muestra el mensaje pero con muchos errores en el formato.	No muestra ningún mensaje.
<b>27. Al finalizar muestra mensaje según formato solicitado en ejemplo</b>	Muestra el mensaje final en el formato correcto y de manera clara.	Muestra el mensaje final correctamente pero con pequeños errores en el formato.	Muestra el mensaje final pero con varios errores en el formato.	Muestra el mensaje final pero con muchos errores en el formato.	No muestra el mensaje final.

#### ESCALA DE NOTAS

Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota
0.0	1.0	10.0	1.5	20.0	1.9	30.0	2.4	40.0	2.9	50.0	3.3	60.0	3.8	70.0	4.4
1.0	1.0	11.0	1.5	21.0	2.0	31.0	2.4	41.0	2.9	51.0	3.4	61.0	3.8	71.0	4.4
2.0	1.1	12.0	1.6	22.0	2.0	32.0	2.5	42.0	2.9	52.0	3.4	62.0	3.9	72.0	4.5
3.0	1.1	13.0	1.6	23.0	2.1	33.0	2.5	43.0	3.0	53.0	3.5	63.0	3.9	73.0	4.6
4.0	1.2	14.0	1.6	24.0	2.1	34.0	2.6	44.0	3.0	54.0	3.5	64.0	4.0	74.0	4.6
5.0	1.2	15.0	1.7	25.0	2.2	35.0	2.6	45.0	3.1	55.0	3.5	65.0	4.0	75.0	4.7
6.0	1.3	16.0	1.7	26.0	2.2	36.0	2.7	46.0	3.1	56.0	3.6	66.0	4.1	76.0	4.8
7.0	1.3	17.0	1.8	27.0	2.3	37.0	2.7	47.0	3.2	57.0	3.6	67.0	4.2	77.0	4.8
8.0	1.4	18.0	1.8	28.0	2.3	38.0	2.8	48.0	3.2	58.0	3.7	68.0	4.2	78.0	4.9
9.0	1.4	19.0	1.9	29.0	2.3	39.0	2.8	49.0	3.3	59.0	3.7	69.0	4.3	79.0	5.0

  

Puntaje	Nota	Puntaje	Nota	Puntaje	Nota
80.0	5.1	90.0	5.8	100.0	6.4
81.0	5.1	91.0	5.8	101.0	6.5
82.0	5.2	92.0	5.9	102.0	6.6
83.0	5.3	93.0	6.0	103.0	6.7
84.0	5.3	94.0	6.0	104.0	6.7
85.0	5.4	95.0	6.1	105.0	6.8
86.0	5.5	96.0	6.2	106.0	6.9
87.0	5.5	97.0	6.2	107.0	6.9
88.0	5.6	98.0	6.3	108.0	7.0
89.0	5.7	99.0	6.4		