

## Tarea 1.1

Fecha de inicio: 6 de Setiembre, 2022  
Fecha de entrega: 11 de Setiembre, 2022

---

Curso: Programación 1 (1100) – Entrega mediante: Gradescope

### Indicaciones generales

1. Recuerda que la tarea es **individual**. Los casos de copia/plagio serán sancionados con nota cero (0) en la asignatura.
2. (a) La fecha límite de entrega es el **domingo 11 de setiembre a las 23:59 hrs.**  
(b) Es altamente recomendable no esperar hasta la última hora.  
(c) *Gradescope* desactivará automáticamente los envíos pasada dicha hora límite.  
(d) **No se aceptarán entregas atrasadas ni entregadas por otros medios.**
3. Revisa bien lo que entregas, aunque en esta oportunidad podrás entregar ilimitadas veces la tarea, la última enviada será la evaluada.
4. Recuerda que *Gradescope* corrige automáticamente tu entrega. Dicha plataforma mostrará si has realizado correctamente las pruebas y mostrará algunos mensajes en color verde. Puedes ver un ejemplo de este caso en el anexo 1.
5. Es posible que hayas subido tu entrega pero hayas modificado algo que no se debió en el template. En ese escenario, *Gradescope* te mostrará algunos mensajes de error. Puedes ver un ejemplo de esto en el anexo 2.

### Gradescope

1. Nosotros les proporcionaremos un código base de donde deberán partir para completar dicho ejercicio. Este archivo es llamado `solution.py` y lo encontrarán en la indicación de la tarea en CANVAS.
2. Al finalizar, **solo** subir el archivo `solution.py` (NO cambiar el nombre del archivo y NO comprimirlo).

3. Cada pregunta tiene casos de prueba. Para obtener la nota completa en una pregunta, el algoritmo debe obtener la respuesta correcta en todos los casos de prueba.
4. Si un caso de prueba falla, visualizarán un mensaje de error con sugerencias. **Lee el error**, revisa el código e inténtalo de nuevo.
5. Los input de los casos de prueba son confidenciales.

### Indicaciones específicas

1. En el anexo 1, se puede ver la plantilla de código del primer ejercicio de esta tarea.
2. Ustedes deben escribir dentro de la sección y a la misma altura de donde esta escrito "*Código comienza aquí*". Además, no deben modificar nada debajo de "*Código acaba aquí*". Recuerden tener cuidado con las indentaciones.
3. Los input del ejercicio se encuentran en la plantilla. Recuerden usar estas variables para resolver el ejercicio.
4. La respuesta del ejercicio debe ser impresa **específicamente** con *print()* para que Gradescope la tome en consideración.
5. Al momento de la impresión de la respuesta, no adicionar texto. Imprimir **única-mente** el resultado que pide el ejercicio.

### Problema 1 - (1 pts)

Leer un nombre  $N$  desde consola e imprimir "Bienvenido {N}!".

#### Ejemplo 1.

**Input :**

```
1 N: Alvaro
```

**Output :**

```
1 Bienvenido Alvaro!
```

### Problema 2 - (1 pts)

Imprimir el número  $N$  elevado al cubo, disminuido en 1.

#### Ejemplo 1.

**Input :**

```
1 N: 3
```

**Output :**

```
1 26
```

**Ejemplo 2.**

**Input :**

```
1 N: -5
```

**Output :**

```
1 -126
```

### Problema 3 - (1 pts)

Leer 2 números enteros  $N$ ,  $M$ . Imprimir  $N^M$ .

**Ejemplo 1.**

**Input :**

```
1 N: 2
```

```
2 M: 3
```

**Output :**

```
1 8
```

### Problema 4 - (1 pts)

Imprimir el número  $N$  dividido entre 2.7182.

**Ejemplo 1.**

**Input :**

```
1 N: 27
```

**Output :**

```
1 9.933043926127585
```

**Problema 5 - (1 pts)**

Imprimir Héroe! si  $N$  es igual a Batman o Superman; caso contrario imprimir Villano!!.

**Ejemplo 1.*****Input*** :

```
1 N: Batman
```

***Output*** :

```
1 Héroe!
```

**Ejemplo 2.*****Input*** :

```
1 N: Joker
```

***Output*** :

```
1 Villano!!
```

**Ejemplo 3.*****Input*** :

```
1 N: Superman
```

***Output*** :

```
1 Héroe!
```

## 1. Anexos

The screenshot displays the 'Autograder Results' page for a student named Fabrizio David Franco Amayo. The page has two tabs: 'Results' (selected) and 'Code'. On the left, a list of test cases is shown, all of which passed. On the right, a summary box shows the student's name, a score of 5.0 / 5.0, and a list of passed tests.

Test Case	Score
Test 1.1 (1.0/1.0)	1.0
Test 1.1 (1.0/1.0)	1.0
Test 1.1 (1.0/1.0)	1.0
Test 1.1 (1.0/1.0)	1.0
Test 1.1 (0.5/0.5)	0.5
Test 1.1 (0.5/0.5)	0.5

**STUDENT**  
Fabrizio David Franco Amayo

**AUTOGRADER SCORE**  
5.0 / 5.0

**PASSED TESTS**  
Test 1.1 (1.0/1.0)  
Test 1.1 (1.0/1.0)  
Test 1.1 (1.0/1.0)  
Test 1.1 (1.0/1.0)  
Test 1.1 (0.5/0.5)  
Test 1.1 (0.5/0.5)

Figure 1: Casos de prueba correctos en Gradescope.

The screenshot displays the 'Autograder Results' page for the same student. The 'Results' tab is selected. The left panel shows the 'Autograder Output (hidden from students)' for a failed test case. The output indicates a file not found error. The right summary box shows a score of 0.0 / 5.0 and a list of failed tests.

**Autograder Output (hidden from students)**

```
cp: cannot stat '/autograder/submit/solution.py': No such file or directory
```

**test\_solution (unittest.loader.\_FailedTest) (0.0/0.0)**

```
Test Failed: Failed to import test module: test_solution
Traceback (most recent call last):
  File "/usr/lib/python3.6/unittest/loader.py", line 428, in _find_test_path
    module = self._get_module_from_name(name)
  File "/usr/lib/python3.6/unittest/loader.py", line 369, in _get_module_from_name
    import _name
  File "/autograder/source/tests/test_solution.py", line 6, in <module>
    from solution import main
ModuleNotFoundError: No module named 'solution'
```

**STUDENT**  
Fabrizio David Franco Amayo

**AUTOGRADER SCORE**  
0.0 / 5.0

**PASSED TESTS**  
test\_solution (unittest.loader.\_FailedTest) (0.0/0.0)

Figure 2: Entrega incorrecta en Gradescope.

```
1
2 # NO MODIFICAR NADA BAJO ESTA LINEA
3 def main():
4 # NO MODIFICAR NADA SOBRE ESTA LINEA
5
6     #=====Pregunta 1=====
7     N = int(input("N: "))
8
9     # Codigo para Pregunta 1 comienza aqui
10
11     print("")
12
13     # Codigo para Pregunta 1 acaba aqui
14
15     #=====Pregunta 2=====
16
17     N = int(input("N: "))
18
19     # Codigo para Pregunta 2 comienza aqui
20
21     print("")
22
23     # Codigo para Pregunta 2 acaba aqui
24     #=====Pregunta 3=====
25
26     N = float(input("N: "))
27
28     # Codigo para Pregunta 3 comienza aqui
29
30     print("")
31
32     # Codigo para Pregunta 3 acaba aqui
33
34
35     #=====Pregunta 4=====
36
37     N = str(input("Nombre: "))
38
39     # Codigo para Pregunta 4 comienza aqui
40
41     print("")
42
43     # Codigo para Pregunta 4 acaba aqui
44
45     #=====Pregunta 5=====
46
47     N = str(input("Animal: "))
```

```
48
49     # Codigo para Pregunta 5 comienza aqui
50
51     print("")
52
53     # Codigo para Pregunta 5 acaba aqui
54
55
56 # NO MODIFICAR NADA BAJO ESTA LINEA
57 if __name__ == '__main__':
58     main()
59 # NO MODIFICAR NADA SOBRE ESTA LINEA
```

Listing 1: Template solution.py.