



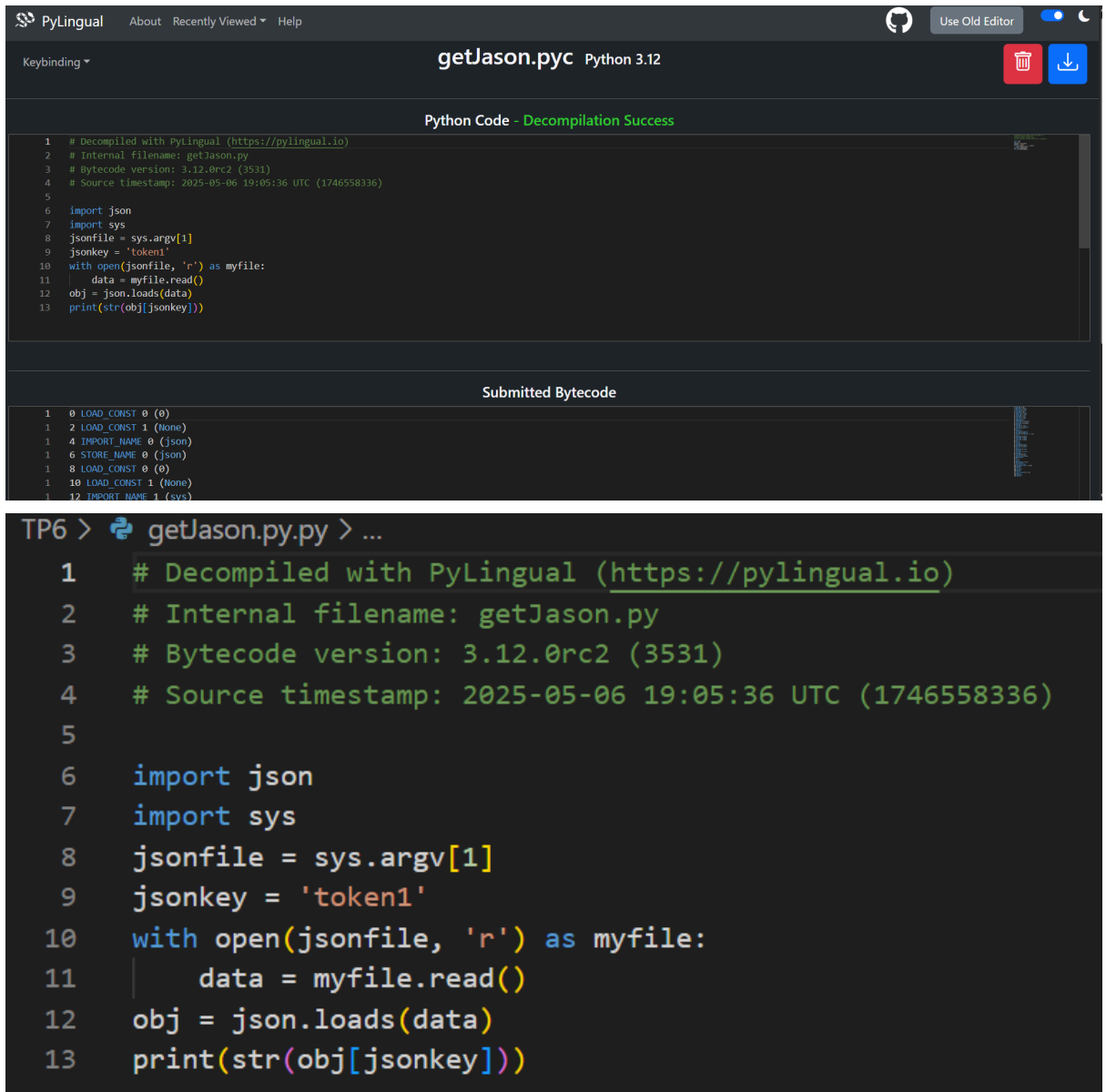
Lic. en Sistemas de Información

Asignatura: Ingeniería en Software 2

Trabajo práctico N° 6

Alumna: Blanchet Ibarra Sofia

1.



The screenshot shows the PyLingual web interface. At the top, there's a header with the PyLingual logo, navigation links (About, Recently Viewed, Help), and a 'Use Old Editor' button. Below the header, the file 'getJason.pyc' is loaded for Python 3.12. The main area displays the decompiled Python code, which includes imports for json and sys, and logic to read a JSON file from a command-line argument. Below the code, there's a section for 'Submitted Bytecode' showing the raw bytecode instructions. At the bottom, a terminal window shows the command 'python getJason.py' being executed, which outputs the decompiled code.

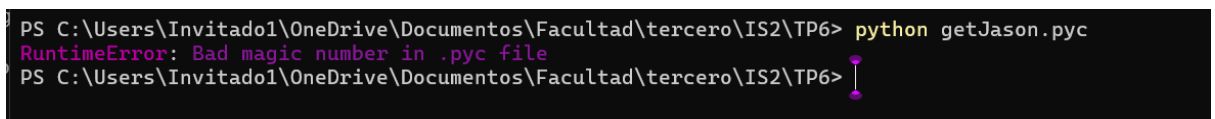
```
1 # Decompiled with PyLingual (https://pylingual.io)
2 # Internal filename: getJason.py
3 # Bytecode version: 3.12.0rc2 (3531)
4 # Source timestamp: 2025-05-06 19:05:36 UTC (1746558336)
5
6 import json
7 import sys
8 jsonfile = sys.argv[1]
9 jsonkey = 'token1'
10 with open(jsonfile, 'r') as myfile:
11     data = myfile.read()
12     obj = json.loads(data)
13     print(str(obj[jsonkey]))
```

```
1 0 LOAD_CONST 0 (0)
1 2 LOAD_CONST 1 (None)
1 4 IMPORT_NAME 0 (json)
1 6 STORE_NAME 0 (json)
1 8 LOAD_CONST 0 (0)
1 10 LOAD_CONST 1 (None)
1 12 IMPORT_NAME 1 (sys)
```

TP6 > `python getJason.py` > ...

```
1 # Decompiled with PyLingual (https://pylingual.io)
2 # Internal filename: getJason.py
3 # Bytecode version: 3.12.0rc2 (3531)
4 # Source timestamp: 2025-05-06 19:05:36 UTC (1746558336)
5
6 import json
7 import sys
8 jsonfile = sys.argv[1]
9 jsonkey = 'token1'
10 with open(jsonfile, 'r') as myfile:
11     data = myfile.read()
12     obj = json.loads(data)
13     print(str(obj[jsonkey]))
```

2.



The screenshot shows a Windows command prompt window. The user has run the command 'python getJason.pyc' in the directory 'C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6'. The output is a 'RuntimeError: Bad magic number in .pyc file', indicating that the file is not a valid Python bytecode file.

```
PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6> python getJason.pyc
RuntimeError: Bad magic number in .pyc file
PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6>
```

```
TP6 > getJason.py.py >...
2 # Internal filename: getJason.py
3 # Bytecode version: 3.12.0rc2 (3531)
4 # Source timestamp: 2025-05-06 19:05:36 UTC (1746558336)
5
6 import json
7 import sys
8 jsonfile = sys.argv[1]
9 jsonkey = 'token1'
10 with open(jsonfile, 'r') as myfile:
11     data = myfile.read()
12     obj = json.loads(data)
13     print(str(obj[jsonkey]))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2> & C:/Users/Invitado1/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Invitado1/OneDrive/Documentos/Facultad/tercero/IS2/TP6/getJason.py.py

Traceback (most recent call last):

File "c:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6\getJason.py.py", line 8, in <module>

jsonfile = sys.argv[1]

~~~~~AAA

IndexError: list index out of range

PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2>

3.

4. Las diferencias entre la documentación y el código reconstruido son las siguientes:
- Siempre usa la clave 'token 1' y no deja pasar otra por línea de comandos, cuando debería permitir elegir cualquier clave del JSON. Si usa 'token1' como valor por defecto, como se esperaba. También toma el nombre del archivo JSON como argumento, pero falla si no se le pasa ninguno, tirando un error en lugar de usar un valor por defecto o mostrar un mensaje claro como nos muestra en la pantalla, directamente se rompe al intentarlo.

5.

```
PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6> python NewgetJason.py sitedata.json
C598-ECF9-F0F7-881A
PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6>
```

```

import json
import sys

# Verifica si se proporcionaron los argumentos necesarios
if len(sys.argv) < 2:
    print("Uso: python getJason.py <archivo_json> [clave]")
    sys.exit(1)

# Primer argumento: nombre del archivo JSON
jsonfile = sys.argv[1]

# Segundo argumento (opcional): clave a buscar en el JSON
# Si no se especifica, se usa 'token1' por defecto
jsonkey = sys.argv[2] if len(sys.argv) >= 3 else 'token1'

# Abre y lee el contenido del archivo JSON
with open(jsonfile, 'r') as myfile:
    data = myfile.read()

# Convierte el texto JSON a un objeto de Python (diccionario)
obj = json.loads(data)

# Busca la clave solicitada dentro del objeto JSON
if jsonkey in obj:
    print(str(obj[jsonkey]))
else:
    print(f"Clave '{jsonkey}' no encontrada en el archivo.")

```

6.

```

PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6> python NewgetJason.py
Uso: python getJason.py <archivo_json> [clave]
PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6>

```

7.

```

PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6> python NewgetJason.py sitedata.json 10
Clave '10' no encontrada en el archivo.
PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6>

PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6> python NewgetJason.py sitedata.json token2
C598-ECF9-F0F7-881B
PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6>

PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6> python NewgetJason.py sitedata.json ClaveSofi
ClavedePrueba1111111
PS C:\Users\Invitado1\OneDrive\Documentos\Facultad\tercero\IS2\TP6>

```