Janet Zhang (QRT)

31st March 2023

1 Installation of packages and Execution of code

1.1 Empty zip file contents on your Desktop.

1.1.1 Open a Command Prompt(CMD) on your desktop. Execute the following lines of code in order.

```
cd Desktop
python3 -m pip install -r requirements.txt
python3
import requests
```

1.1.2 Open a 2nd Command Prompt(CMD) on your desktop to check connection status. Execute the following lines of code in order.

```
cd Desktop
python3 -m app
```

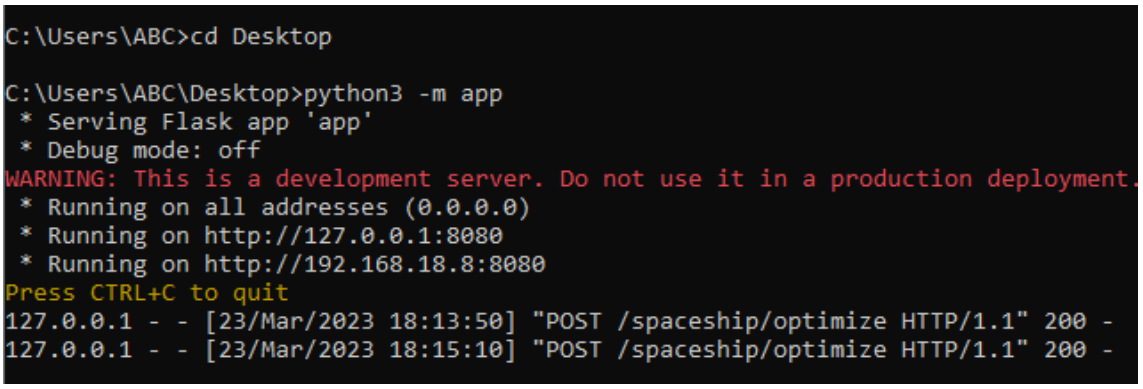
1.1.3 Back to the first Command Prompt. Execute the following lines of code in order.

```
response = requests.post("http://127.0.0.1:8080/spaceship/optimize",json=[{"name":
"Contract1", "start": 0, "duration": 5, "price": 10},{"name": "Contract2",
"start": 3, "duration": 7, "price": 14},{"name": "Contract3", "start": 5,
"duration": 9, "price": 8},{"name": "Contract4", "start": 5, "duration": 9,
"price": 7}])
response.json()
```

```
C:\Users\ABC\Desktop>python3 -m pip install -r requirements.txt
Requirement already satisfied: requests==2.28.2 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation.p
ython.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from -r requirements.txt (line 1)) (2
.28.2)
Requirement already satisfied: flask==2.2.3 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation.pytho
n.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from -r requirements.txt (line 2)) (2.2.3
)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\abc\appdata\local\packages\pythonsoftwarefoun
dation.python.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from requests==2.28.2->-r req
uirements.txt (line 1)) (3.1.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundat
ion.python.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from requests==2.28.2->-r requir
ements.txt (line 1)) (1.26.15)
Requirement already satisfied: idna<4,>=2.5 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation.pytho
n.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from requests==2.28.2->-r requirements.tx
t (line 1)) (3.4)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation
.python.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from requests==2.28.2->-r requireme
nts.txt (line 1)) (2022.12.7)
Requirement already satisfied: Jinja2>=3.0 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation.python
.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from flask==2.2.3->-r requirements.txt (li
ne 2)) (3.1.2)
Requirement already satisfied: itsdangerous>=2.0 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation.
python.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from flask==2.2.3->-r requirements.t
xt (line 2)) (2.1.2)
Requirement already satisfied: Werkzeug>=2.2.2 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation.py
thon.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from flask==2.2.3->-r requirements.txt
 (line 2)) (2.2.3)
Requirement already satisfied: click>=8.0 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation.python.
3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from flask==2.2.3->-r requirements.txt (lin
e 2)) (8.1.3)
Requirement already satisfied: colorama in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation.python.3.
10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from click>=8.0->flask==2.2.3->-r requirement
s.txt (line 2)) (0.4.6)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\abc\appdata\local\packages\pythonsoftwarefoundation.py
thon.3.10_qbz5n2kfra8p0\localcache\local-packages\python310\site-packages (from Jinja2>=3.0->flask==2.2.3->-r req
uirements.txt (line 2)) (2.1.2)

[notice] A new release of pip available: 22.3.1 -> 23.0.1
[notice] To update, run: C:\Users\ABC\AppData\Local\Microsoft\WindowsApps\PythonSoftwareFoundation.Python.3.10_qb
z5n2kfra8p0\python.exe -m pip install --upgrade pip

C:\Users\ABC\Desktop>python3
Python 3.10.10 (tags/v3.10.10:aad5f6a, Feb  7 2023, 17:20:36) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import requests
>>> response = requests.post("http://127.0.0.1:8080/spaceship/optimize", json=[{"name": "Contract1", "start": 0,
"duration": 5, "price": 10}, {"name": "Contract2", "start": 3, "duration": 7, "price": 14}, {"name": "Contract3",
 "start": 5, "duration": 9, "price": 8},{"name": "Contract4", "start": 5, "duration": 9, "price": 7}])
>>> response.json()
{'income': 18, 'path': ['Contract1', 'Contract3']}
```



1.1.4 The HTTP 200 OK success status response code indicates that the request has succeeded.

2 Introduction to Code

2.1 This code defines a Flask web server that listens on port 8080. When a POST request is received at the endpoint /spaceship/optimize with a JSON payload containing the contracts list, it calls the findOptimalContracts function to compute the maximum profit and returns a JSON response.

2.2 The Python function findOptimalContracts takes a list of contracts as input and returns a dictionary with two keys: "income" and "path". The "income" key corresponds to the maximum total profit that can be obtained by selecting a subset of contracts, and the "path" key corresponds to the sequence of contract names that lead to this maximum profit.

2.2.1 The contracts are first sorted in ascending order by their "start" time using the sorted function and a lambda function as a key.

2.2.2 The function then initializes a dynamic programming table dp with n rows, where n is the number of contracts. Each row is a dictionary with two keys: "price" and "prev". The "price" key corresponds to the maximum total profit that can be obtained by selecting a subset of contracts that includes the contract at that row, and the "prev" key corresponds to the index of the previous contract in the optimal sequence.

2.2.3 The function then iterates over each contract in the sorted list and computes the maximum total profit that can be obtained by selecting a subset of contracts that includes that contract. This is done by iterating over all previous contracts and checking if adding the current contract to the subset is feasible (i.e., if the end time of the previous contract is before the start time of the current contract). If it is feasible and the total profit of the subset including the previous contract is greater than the total profit of the subset including the current contract, then the "price" and "prev" keys of the current row in the dynamic programming table are updated with the values of the previous row.

2.2.4 Finally, the function finds the row with the maximum total profit and constructs the optimal contract sequence by following the "prev" keys from that row to the beginning of the sequence. The sequence is then reversed and returned along with the maximum total profit as a dictionary with the keys "income" and "path".