

Yifu He  
190003956

MC answers

Questions	Answer
1	B
2	A
3	B
4	A
5	C
6	C
7	B
8	A
9	D
10	D
11	D
12	A
13	C
14	A
15	D
16	A
17	B
18	A
19	A
20	B
21	D
22	B
23	A
24	A
25	C
26	B
27	B
28	D
29	D
30	D

SA1:

```
3]: # SA1
def prefix_sum(arr):
    for i in range(1, len(arr)):
        arr[i] += arr[i-1]
    return arr
a = [1, 3, 5, 7, 2, 4, 6, 8]
print(prefix_sum(a))
```

3]: [1, 4, 9, 16, 18, 22, 28, 36]

SA2:

```
|: # SA2
[(i, i*i) for i in range(5)]
|: [(0, 0), (1, 1), (2, 4), (3, 9), (4, 16)]
```

SA3:

```
: # SA3
# a n
# b nlog(n)
```

- a) n
- b)  $n \log(n)$

SA4:

```
| def upique_and_sorted(s):
|     d = {}
|     for i in s:
|         if i not in d:
|             d[i]=0
|     res = list(d.keys())
|     res.sort()
|     return res
upique_and_sorted([1,3,5,3,4,6,7,8,2,3,4])
|
| [1, 2, 3, 4, 5, 6, 7, 8]
```

SA5:

```
| # SA5
| def dict1(func):
|     D = {0:0,1:1,2:2}
|     def _decorator(param):
|         if param in D:
|             return D[param]
|         else:
|             D[param] = _decorator(param-1) + _decorator(param - 2) + _decorator(param//2)
|             return D[param]
|     return _decorator
@dict1
def f(n):
    if n ==0:
        return 0
    elif n == 1:
        return 1
    elif n == 2:
        return 2
    else:
        return f(n-1) + f(n-2)+f(n//2)
print(f(14))
print(f(200))

1664
1317261742621463123234288924400109075146188
```

SA6:

```
: # SA6
# He ended his speech saying 'Your team completed the project beautifully'
def q6():
    temp = input("Enter a sentence: ")
    L = temp.split(" ")
    res = []
    for i in L:
        if i[-2:] == "ed":
            res.append(i)
    print("The word ended with ed are here")
    for ele in res:
        print(ele)
q6()
```

```
Enter a sentence: He ended his speech saying 'Your team completed the project beautifully'
The word ended with ed are here
ended
completed
```

---

SA7:

---

```
: # SA7
print("The 2 by 3 array is ")
res = np.array([2**i for i in range(6)]).reshape(2,3)
print(res)
print("The flatten array is")
print(res.reshape(-1))
```

```
The 2 by 3 array is
[[ 1  2  4]
 [ 8 16 32]]
The flatten array is
[ 1  2  4  8 16 32]
```

---

SA8:

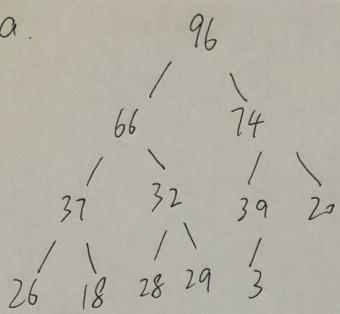
---

```
|: def q8(sentence):
|:     L = sentence.split(" ")
|:     res = {}
|:     for i in L:
|:         i = i.lower()
|:         if i in res:
|:             res[i] +=1
|:         else:
|:             res[i] = 1
|:     print("WORD\t\tCOUNT")
|:     for key,value in res.items():
|:         if value >=2:
|:             print(key, "\t\t", value)
q8("this is sample text with several words this is more sample text with some different words")
```

WORD	COUNT
this	2
is	2
sample	2
text	2
with	2
words	2

Q1:

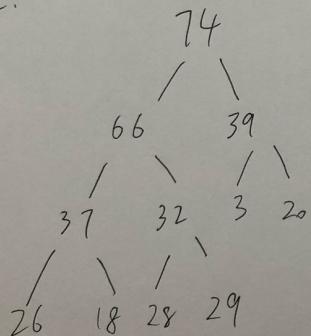
Q1.a.



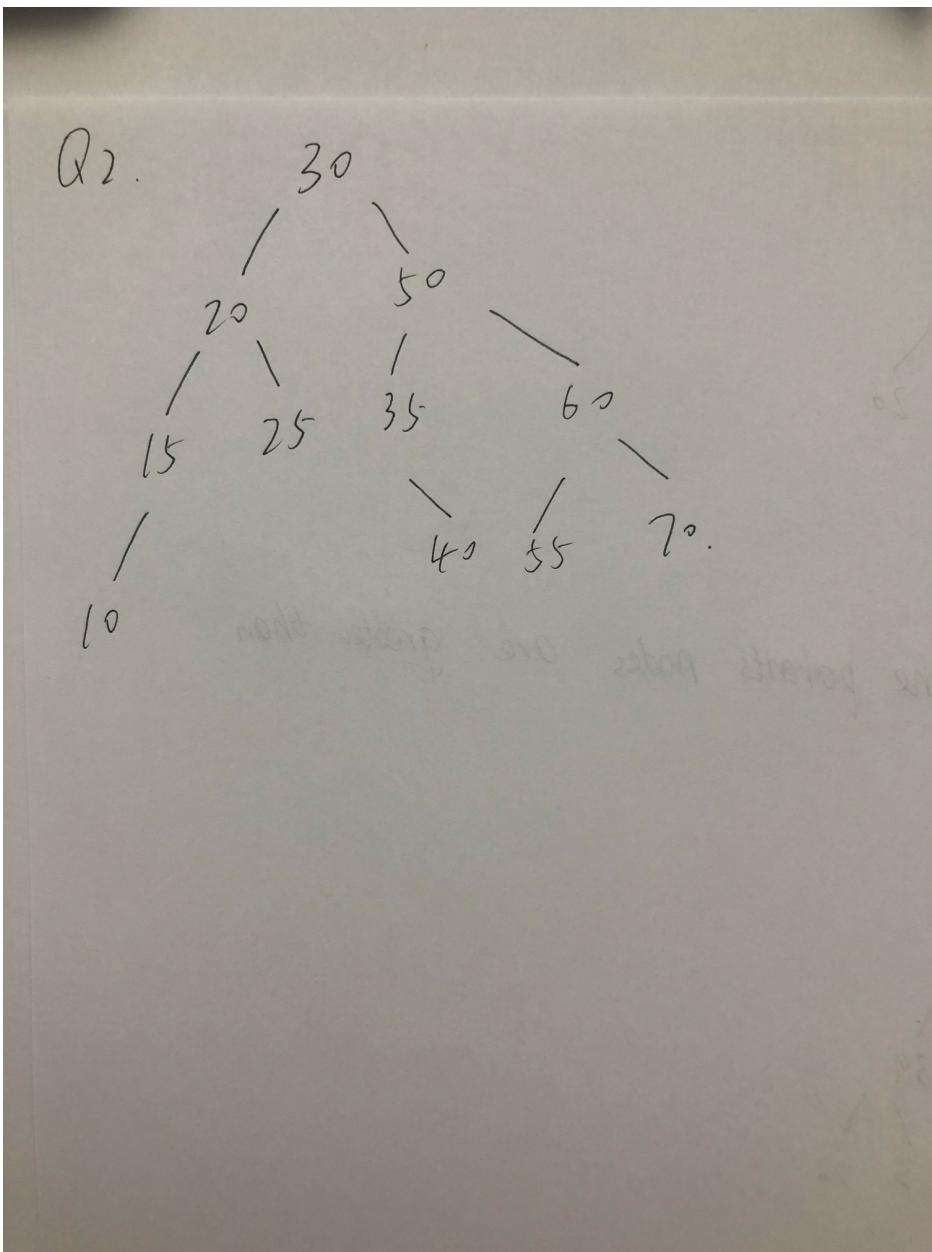
Q1.b

because all the parents nodes are greater than  
child nodes

Q1.c.



Q2:



Q3:

```
class Sentence(object):
    def __init__(self,sentence):
        self.sentence = sentence.split(" ")

    def get_first_word(self):
        return self.sentence[0]

    def get_last_word(self):
        return self.sentence[-1]

    def get_all_words(self):
        return self.sentence

    def replace(self,index,newWord):
        if index >=len(self.sentence):
            print("Error in replace: index out of range")
        else:
            self.sentence[index] = newWord
```

```
a = Sentence("I am the great guy")
a.get_first_word()
```

```
'I'
```

```
a.get_all_words()
```

```
['I', 'am', 'the', 'great', 'guy']
```

```
a.replace(3,"handsome")
a.get_all_words()
```

```
['I', 'am', 'the', 'handsome', 'guy']
```

```
a.get_last_word()
```

```
'guy'
```

```
a.replace(10,"pretty")
```

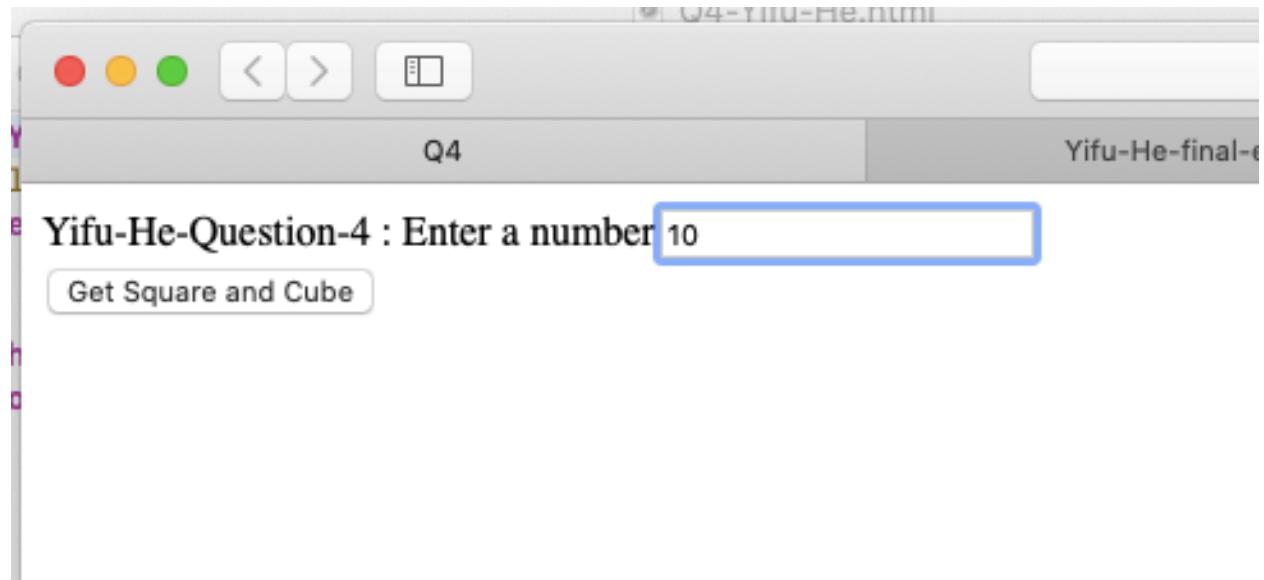
```
Error in replace: index out of range
```

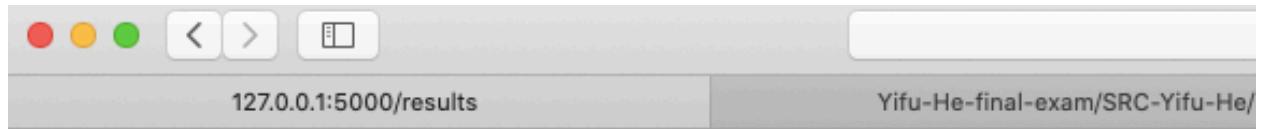
```
a.replace(-3,"pretty")
```

```
Error in replace: index out of range
```

Q4:

Result:





### Number Square Cube

1	1	1
2	4	8
3	9	27
4	16	64
5	25	125
6	36	216
7	49	343
8	64	512
9	81	729
10	100	1000

code:

```
: from flask import Flask
from flask import render_template
from flask import request

app = Flask(__name__)

@app.route("/")
def root():
    return render_template('Q4-Yifu-He.html')

@app.route("/results", methods=['POST'])
def results():
    n = request.form['number']
    html = f"<h1>Output Square and cube from 1 to {n}</h1><table><tr> <th>Number</th><th>Square</th><th>Cube</th></tr>"
    for i in range(1,int(n)+1):
        html += f"<tr><td>{str(i)}</td><td>{str(i*i)}</td><td>{str(i**3)}</td></tr>"
    return html
if __name__ == '__main__':
    app.run()
```

HTML:

Q4-Yifu-He.html

Q4-Yifu-He.html No Selection

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <title>Q4</title>
6   </head>
7   <body>
8     <form action="/results" method="post">
9       <label>Yifu-He-Question-4 : Enter a number</label><input type="text"
10      name="number" id="number"><br>
11      <button type="submit">Get Square and Cube</button>
12    </form>
13  </body>
14 </html>
```

Q5:

Code:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import datetime
_dt = datetime.datetime
# define the function to change the format of date
def format_time(str_obj):
    return datetime.datetime.strptime(str_obj,"%m/%d/%Y %H:%M")
```

```
path = "/Users/yifuhe/Desktop/Yifu-He-final-exam/stock.csv"
data = pd.read_csv(path)
start = data[data["date"] == "2/2/2009 0:00"].index[0]
end = data[data["date"] == "6/30/2010 0:00"].index[0]
data = data.loc[start:end,:].reset_index().drop("index",axis=1)
data["date"] = data["date"].apply(format_time)
data["20 days"] = data["AA"].rolling(20).mean()
data["50 days"] = data["AA"].rolling(50).mean()
data["200 days"] = data["AA"].rolling(200).mean()
new_data = data.rename(columns={"AA": "Daily"})
new_data

plt.plot(data["date"], new_data["Daily"], color = "black", label="Daily")
plt.plot(data["date"], data["20 days"], color = "red", label="20 days")
plt.plot(data["date"], data["50 days"], color = "green", label = "50 days")
plt.plot(data["date"], data["200 days"], color = "blue", label="200 days")
plt.title("AA Stock")
plt.xlabel("date")
plt.ylabel("Price")
plt.legend(loc = "upper left")
my_x_ticks = [_dt(2009,i,1) for i in range(3,12,2)] + [_dt(2010,i,1) for i in range(1,6,2)]
["2019.03","2019.05","2019.07","2019.11","2010.01","2010.03","2010.05"]
plt.xticks(my_x_ticks, rotation = 45)
# plt.xlim(("2019.03","2010.05"))
plt.show()

/Users/yifuhe/opt/anaconda3/lib/python3.7/site-packages/pandas/plotting/_matplotlib/converter.py
```

Plot:

