

## Bài tập môn kĩ thuật lập trình chương 3

Bài 1: Rewrite your pay computation to give the employee 1.5 times the hourly rate for hours worked above 40 hours.

The screenshot shows the VS Code interface with the code for Exercise 1 in the editor and its execution in the terminal below.

```
bài 1 c3.py > ...
1 Hours = float(input("Enter hours"))
2 Rate = float(input("Enter Rate"))
3 Standar_hour=40
4 if Hours <= Standar_hour:
5     Pay= Hours*Rate
6 else:
7     Standar_pay = Standar_hour * Rate
8     Overtime_hours= Hours-Standar_hour
9     Overtime_pay= overtime_hours*Rate*1.5
10    Pay= Standar_pay+overtime_pay
11 print("Pay: ", Pay)
12
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe "d:/VS_CODE/bài 1 c3.py"
Enter hours41
Enter Rate2
Pay: 83.0
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe "d:/VS_CODE/bài 1 c3.py"
Enter hours45
Enter Rate10
Pay: 475.0
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe "d:/VS_CODE/bài 1 c3.py"
Enter hours[]
```

Bài 2: Rewrite your pay program using try and except so that your program handles non-numeric input gracefully by printing a message and exiting the program. The following shows two executions of the program:

The screenshot shows the VS Code interface with the code for Exercise 2 in the editor and its execution in the terminal below.

```
baif2c3.py > ...
1 try:
2     hours = float(input("Enter Hours: "))
3     rate = float(input("Enter Rate: "))
4 except:
5     print("Error, please enter numeric input")
6 else:
7     if hours > 40:
8         pay = 40 * rate + (hours - 40) * rate * 1.5
9     else:
10        pay = hours * rate
11 print("Pay:", pay)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe d:/VS_CODE/baif2c3.py
Enter Hours: tri
Error, please enter numeric input
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe d:/VS_CODE/baif2c3.py
Enter Hours: 20
Enter Rate: nine
Error, please enter numeric input
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe d:/VS_CODE/baif2c3.py
Enter Hours: forty
Error, please enter numeric input
```

Bài 3: Write a program to prompt for a score between 0.0 and 1.0.

If the score is out of range, print an error message.

If the score is between 0.0 and 1.0, print a grade using the following table:

```
❸ bài 3c3.py > ...
1  score = input("Enter score: ")
2
3  try:
4      score = float(score)
5  except:
6      print("Bad score")
7  else:
8      if score < 0.0 or score > 1.0:
9          print("Bad score")
10     elif score >= 0.9:
11         print("A")
12     elif score >= 0.8:
13         print("B")
14     elif score >= 0.7:
15         print("C")
16     elif score >= 0.6:
17         print("D")
18     else:
19         print("F")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe "d:/VS_CODE/bài 3c3.py"
Enter score: 0.95
A
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe "d:/VS_CODE/bài 3c3.py"
Enter score: perfect
Bad score
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe "d:/VS_CODE/bài 3c3.py"
Enter score: 10.0
Bad score
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe "d:/VS_CODE/bài 3c3.py"
Enter score: 0.75
C
PS D:\VS_CODE> & C:/Users/tuant/AppData/Local/Programs/Python/Python313/python.exe "d:/VS_CODE/bài 3c3.py"
Enter score: 0.5
F
```