



PROFICIENCY EXAM

COMP1010 Introduction to Programming

Sep. 29, 2021

Issued: 04:00 PM – Due: 5:50 PM

Exam Submission Instructions:

- You should follow “Proficiency Exam Setup” before starting the exam. Students who do not meet the setup requirements will not be able to receive results.
- There are 3 Problems for a total of 100 points. Each problem may contain several questions.
- Your program should work correctly on all inputs. If there are any specifications about how the program should be written (or how the output should appear), those specifications should be followed.
- Your code and functions/modules should be appropriately commented. However, try to avoid making your code overly busy (e.g., include a comment on every line).
- Variables and functions should have meaningful names, and code should be organized into functions/methods where appropriate.
- Academic honesty is required in all work you submit to be graded. You should **NOT** copy or share your code with other students to avoid plagiarism issues.
- **This is closed-book exam, and no Internet access is allowed (except for Canvas submission). Use, e.g, `help(print)` to check the built-in function usage if necessary. Drafting paper and pen/pencil are allowed. DO NOT USE any web-based drafting/-drawing tools during the exam.**
- You can use all the built-in functions, package, knowledge you know to solve these problems (must be in Python).
- **You should upload your .py file(s) to the Canvas before the end of the exam session (5.50PM, Sep. 29, 2021).** Submit .py file with the following naming format: **YourStudentID_ProExam_Q1.py**, for example: **V202000999_ProExam_Q1.py**. Note: If you are working on Jupiter Notebook, you need to download/convert it to Python .py file for submission.

Problem 1 (30-pts)

- (a) **Flat Tax: (10-pts)** A flat tax (short for flat-rate tax) is a tax system where the tax payment is calculated as fixed percentage of income. For example, with a tax rate of 20%, if income is \$10,000 then the tax due is $20\% \times \$10,000 = \$2,000$ and if income is \$55,000 then tax due is \$11,000. (Source: Wikipedia)

Write a program that prompt user for his or her income, a flat tax rate (a number between 1-100), and print out the tax due amount as shown in Figure 1.

```
Enter Your Annual Income: 10000
Enter your flat tax-rate (%): 15
Your tax due amount is: $1,500.00
```

Figure 1: Example test case for Part (a).

- (b) **Progressive Tax: (20-pts)** A progressive tax is a tax in which the tax rate increases as the taxable amount increases. The term progressive refers to the way the tax rate progresses from low to high, with the result that a taxpayer's average tax rate is less than the person's marginal tax rate (Source Wikipedia).

A simplified Viet Nam tax Brackets is shown in Figure 2. An example of the tax amount due for an income of \$10,000 is calculated as below:

$$5/100 * 2,500 + 10/100 * (5,000 - 2,500) + 18/100 * (10,000 - 5,000) = 1,275$$

Hence, the effective (average) tax rate is $1,275/10,000 = 12,75\%$

Simplified Viet Nam Tax Brackets		
No	Tax Rate %	Annual Taxable Income (USD)
1	5%	Up to \$2,500
2	10%	From \$2,501 to \$5,000
3	18%	From \$5,001 to \$16,500
4	28%	From \$16,501 to \$41,500
5	35%	More than \$41,500

Figure 2: Simplified Viet Nam tax brackets in 2019.

Write a program that prompt user for his or her annual income and

- (i) **(10-pts)** Print out the amount tax due (write a function, name it as `progressive_tax()` with one input is income amount, and output is tax due amount).
- (ii) **(10-pts)** Print out the effective tax rate.

Note: The print out number need to be formatted exactly as appeared in the examples, otherwise up to 5-points will be deducted.

```
Enter Your Annual Income: 2000
Your tax due amount is: $100.00
Your effective tax rate is: 5.00%

Enter Your Annual Income: 4000
Your tax due amount is: $275.00
Your effective tax rate is: 6.88%

Enter Your Annual Income: 10000
Your tax due amount is: $1,275.00
Your effective tax rate is: 12.75%

Enter Your Annual Income: 33000
Your tax due amount is: $7,065.00
Your effective tax rate is: 21.41%

Enter Your Annual Income: 55000
Your tax due amount is: $14,170.00
Your effective tax rate is: 25.76%
```

Figure 3: Example test case for Part (b).

Problem 2 (40-pts)

- (a) **(20-pts)** Prompt user for an integer greater than or equal to 2, repeat if the input value is less than 2. Next, using the `for-loop` to print out all the possible pairs of two positive integers whose sum are equal to this number. (See example output).
- (b) **(20-pts) [Hard]** Using the `recursion` method to print out all possible sets of multiple positive integers whose sum are equal to this number.

Hint: Loop through all possible pairs of two numbers whose sum equals `n`, and recursion on the first (or second) term while keeping track of the current sum state in a string.

Note that you should follow the printout format, but the order does not matter (also, it is also okay to print out more than needed but not less (for example, `5=1+4+` is acceptable). Please use a single Python file for this problem.

```
Enter an integer greater than or equal to 2: 1
Enter an integer greater than or equal to 2: 4

a) All pairs of two positive integers whose sum are equal to 4:
3+1
2+2
1+3

b) All sets of multiple positive integers whose sum are equal to 4:
4
3+1
2+2
1+2+1
1+3
2+1+1
1+1+2
1+1+1+1
```

Figure 4: Example A for Problem 2.

```
Enter an integer greater than or equal to 2: 5

a) All pairs of two positive integers whose sum are equal to 5:
4+1
3+2
2+3
1+4

b) All sets of multiple positive integers whose sum are equal to 5:
5
4+1
3+2
1+3+1
2+3
2+2+1
1+2+2
1+1+2+1
1+4
3+1+1
2+1+2
1+2+1+1
1+1+3
2+1+1+1
1+1+1+2
1+1+1+1+1
```

Figure 5: Example B for Problem 2.

Problem 3 (30-pts)

VinUniversity plans to establish a new research center in 2022. It is interested in building an automatic system for calculating weekly pay based on the employee's level and number of hours worked. Specifically, undergraduates students earn \$15/hour. Postdoctoral researchers and staffs earn \$25/hour and \$35/hour, respectively. Note that undergraduates students cannot be paid for more than 20 hours in one week. Thus, if they have worked more than 20 hours in a week, then they will only be paid for the first 20 hours. Postdoctoral researchers and staff have no limit on the number of hours that they can be paid for.

Create the function called "vinuniPayment" that calculates and returns the total pay. It has two parameters: **level** and **hours**, in which level is a string representing the classification level of each research center member (ugrad is undergraduate, post is postdoctoral researcher, and staff is staff member); and **hours** is a float representing the number of hours worked.

Once the function has been created, call the following scripts:

```
vinuniPayment('ugrad', 24.0)
vinuniPayment('staff', 35.5)
vinuniPayment('post', 10.5)
```