





Assignment 4

Due Date for Programs: October 5, 2016



The parts of this assignment marked with the "alone" icon, , should be completed individually by each student.

You may discuss any parts of the assignments with your classmates and tutors (or anyone else) but you are responsible for understanding all of the parts of the assignment that you submit. Any sharing or copying of the parts of the assignments marked with "alone" icon, , will be considered cheating.

You should not use any features of Python that have not been covered in class or recitations. If you have doubt if you are allowed to use certain structures, just ask one of the instructors.

For this assignment you will be writing a few programs which should be saved independently as their own ".py" files. The filename you should use for each program is outlined in the sections below. When you're finished you should submit your programs to the Assignment 4 category inside of NYU Classes.



Problem 1 (20 points): Letter Grade

Translate a numeric grade to a letter grade. (This problem does not require you to compute any average grades.) The program should ask the user for a numeric grade. It should then use the table below to calculate the corresponding letter:

90-100	-	A
80-89	-	B
70-79	-	C
60-69	-	D
0-59	-	F

(Note that we are omitting all of the + and - versions of the grades to keep the program less tedious.)

The program should print out both the number and letter grade. If the value entered by the user is not numeric, your program will terminate with an error - that's fine. However, if the number is not within the ranges specified in the table, the program should output: "Could not translate NUM into a letter grade" where NUM is replaced by the value entered by the user.

Comment your source code 1) briefly describing parts of your program 2) include your name, the date, and your class section at top of your file (above everything else)

Here are a few sample runs of the program:

Output:

```
What grade did you get? 59
Number Grade: 59
Letter Grade: F
```

Output:

```
What grade did you get? 86.5
Number Grade: 86.5
Letter Grade: B
```

Output:

```
What grade did you get? -12
Couldn't translate -12 into a letter grade
```

What to Submit

This program should be named (i.e., the name of the file containing the program should be) `letter_grade.py`. You only need to submit the source code for this problem.



Problem 2 (25 points): Market Basket

Category	Weight (%)	2006-2007 change (%)
Food and beverages	15.009	4.0
Housing	44.377	3.0
Apparel	3.697	-1.5
Transportation	16.030	1.2
Medical care	5.780	4.8
Recreation	5.387	-0.4
Education and communication	6.455	2.7
Other goods and services	3.265	2.6
All items	100	2.58

The above table shows the market basket for an average American in years 2006-2007. For that average basket, the inflation was 2.58.

Write a program that calculates a personal rate of inflation for an individual. Your program should prompt the user for their own weights associated with each of the eight categories. Your program should assume percentage price change to be exactly what the table above shows.

Calculate the personal inflation rate based on that information. In addition, display a message indicating if the personal inflation rate was higher or lower than the one shown in the table.

Comment your source code 1) briefly describing parts of your program 2) include your name, the date, and your class section at top of your file (above everything else)

Here is a sample run of the program:

Output:

```
Enter your personal rates for each category. (Unassigned percentage will be automatically placed in
"Other" category

Food and beverage: 20
Housing: 25
Apparel: 4
Transportation: 10
Medical care: 2
Recreation: 15
Education: 25

-----
-----
Your personal inflationrate, 2.295, is lower than the national average.
-----
-----
```

Extra credit: For additional 5 points, modify the program to display the table containing the market basket in the above image. If you do that, a sample run of the program should look something like the following

Output:

Category	Weight (%)	2006-2007 change (%)
Food and beverage	15.009	4.0
Housing	44.377	3.0
Apparel	3.697	-1.5
Transportation	16.030	1.2
Medical care	5.780	4.8
Recreation	0.539	-0.4
Education and communication	6.455	2.7
Other goods and services	3.265	2.6
All items	100.000	2.603



Enter your personal rates for each category. (Unassigned percentage will be automatically placed in "Other" category)

Food and beverage: 25
Housing: 0
Apparel: 10
Transportation: 5
Medical care: 2
Recreation: 20
Education: 10

Your personal inflationrate, 1.924 , is lower than the national average.

What to Submit

This program should be named (i.e., the name of the file containing the program should be) `market_basket.py`. You only need to submit the source code for this problem.



Problem 3 (25 points): Improving GPA

Write a program that calculates the minimum GPA that a student needs to earn in the current semester to reach their target GPA. If this is possible, print the GPA for this semester that the student should be aiming for. If it is not possible, print a message stating that this is not a possibility.

Prompt the user for his/her current GPA.

Prompt the user for the number of credits he/she completed.

Prompt the user for the number of credits he/she is currently taking.

Prompt the user for the value of their target GPA at the end of the current semester.

Use the algebraic formula that we derived in class (or for extra practice derive it yourself).

Comment your source code 1) briefly describing parts of your program 2) include your name, the date, and your class section at top of your file (above everything else)

Here is a sample run of the program:

Output:

```
Enter your current GPA: 3.4
Enter the number of creadits you earned so far: 80
Enter the number of creadits you are taking this semester: 16
What is your target GPA: 3.5
You need to earn 4.0 this semester
```

Output:

```
Enter your current GPA: 2.0
Enter the number of creadits you earned so far: 30
Enter the number of creadits you are taking this semester: 15
What is your target GPA: 3.0
Your target GPA is not possible to achieve.
```

What to Submit

This program should be named (i.e., the name of the file containing the program should be) `improve_gpa.py`. You only need to submit the source code for this problem.



Problem 4 (30 points): Calculating Taxes - Take 2

Last week, you were looking at the tax brackets for years 2015 and 2016 to figure out the taxes that a person would pay. Most of you arrived at a conclusion that if one gets an increase that moves him/her to the hire bracket, then the take home income actually decreases.

This is not quiet correct. That's because the taxes are not computed by just taking a single percentage of the total income. Read about a more realistic way of computing taxes at at <http://blog.taxact.com/how-tax-brackets-work/>.

Use the procedure described above to write a program that calculates the taxes for an individual (assume that the tax is based purely on the tax bracket into which they fall) as well as their "take home" amount after the taxes.

Prompt the user for his/her annual salary.

Use the table at <http://taxfoundation.org/article/2016-tax-brackets> to determine which tax bracket is applicable (your program should be able to handle salaries that come from any of the brackets).

Print the information about the tax amount and take home amount.

Comment your source code 1) briefly describing parts of your program 2) include your name, the date, and your class section at top of your file (above everything else)

Here is a sample run of the program:

Output:

```
Enter your salary: 50000
Your taxes for 2016 on $50,000.00 are $8,271.25
Your take-home income is $41,728.75
```

Output:

```
Enter your salary: 150000
Your taxes for 2016 on $150,000.00 are $35,036.75
Your take-home income is $114,963.25
```

Output:

```
Enter your salary: 500000
Your taxes for 2016 on $500,000.00 are $154,169.95
Your take-home income is $345,830.05
```

What to Submit

This program should be named (i.e., the name of the file containing the program should be) `taxes.py`. You only need to submit the source code for this problem.

Note: the TaxAct actually has exactly the tool that you are building on their website. You can verify your own number there: <https://www.taxact.com/tools/tax-bracket-calculator.asp> - just make sure to select 2016 as the year.

Grading

The only way to receive the credit for the worksheet problems is to hand them in before the end of the lab session in which they are given.

The programs are graded based on correctness, style of code, design and documentation.

What and how to submit?

You should submit the source code file for each program to NYU Classes by the due date stated above. Make sure that you get an email confirmation after you submit the assignment. You should keep that email until the grades are returned - it is your proof that the assignment was submitted! If you do not get an email confirmation, you should try to resubmit the assignment. If you do not get that email, it means that we did not get your assignment.