Lone Star
section 4

Branching quiz

1. (3.1) Use the following code to answer the below questions

```
1  if my_var % 2 == 1:
2    if my_var**3 != 27:
3      my_var = my_var +4  #Assignment 1
4    else:
5      my_var /= 1.5       #Assignment 2
6  else:
7    if my_var <= 10:
8      my_var *= 2         #Assignment 3
9    else:
10     my_var -= 2         #Assignment 4
11 print(my_var)
```

   (a) Find four values of my_var so each of the four assignment statements will be executed: each value should cause one assignment statement to be executed.

   (b) Find four ranges of my_var values that will cause each of the four assignment statements to be executed.

2. (3.2) In Harry Potter, the currency consists of knuts, sickle, and galleon. There are 29 knuts in one sickle and 17 sickles in one galleon. Write a program that will convert some amount of knuts into the fewest amount of coins possible. Only print non-zero values, meaning don't print something similar to "0 sickles." For example,

   - Given 32 knuts, output 1 sickle 3 knuts
   - Given 544 knuts, output 1 galleon 4 sickles 18 knuts
   - Given 993 knuts, output 2 galleons 7 knuts. Do **not** output 2 galleons 0 sickle 7 knuts.

3. (3.3) The table below shows the maximum health of characters based on race and class for a new video game I am creating. Write a program that asks the user for the race and the class of their character, and then sets the *health_points* variable according to the table below.

health_points = -1
#Your code here.

	Race	
Class	Elf	Ogre
Warrior	150	200
Bard	75	100
Wizard	25	50

Dot Matrix
section 5

Branching quiz

1. (3.1) Write a program that prompts the user for a letter and checks whether the letter is a vowel or consonant. A vowel should output "*vowel*", and a consonant should output *consonant*. You may assume only lower case letters. Below is sample output.
   Hint: In the English language, a, e, i, o, and u are the vowels.

```
Enter a letter: a
vowel
```

```
Enter a letter: x
consonant
```

2. (3.2) Create a game of Rock, Paper, Scissors that takes user inputs. The first input should be player 1 and the second input should be player 2. Print the winner according to the following rules.

   - Rock beats Scissors
   - Scissors beats Paper
   - Paper beats Rock

   For example:

```
Player 1 choice: Rock
Player 2 choice: Paper
player 2 wins!
```

```
Player 1 choice: Scissors
Player 2 choice: Paper
player 1 wins!
```

```
Player 1 choice: Rock
Player 2 choice: Rock
It's a tie!
```

3. (3.3) The table below show what your resting heart rate should be based on age and athleticism. Write a program that asks the user their age and desired athleticism goal, and then outputs what their resting heart rate should be.

	Athleticism	
Age	Above Average	Below Average
20 – 39	47 – 72	73 – 93
40 – 59	46 – 71	72 – 94
60 – 79	45 – 70	71 – 97

Your end output should look similar to this

```
Enter your age: 45
Enter your athleticism goal: Below Average
Your resting heart rate should be between 72–94.
```

Dark Helmet
section 4

Branching quiz

1. (3.1) Write a program that asks the user for three numbers, and then determines (and outputs) which of the numbers is the smallest. Do not use the built-in function *min*().
   For example,

```
Pick a number: 35
Pick another number: 11
Pick another number: 89
The smallest number is 11.
```

```
Pick a number: 3
Pick another number: 2
Pick another number: 1
The smallest number is 1.
```

2. (3.2) Primary U.S. interstate highways are numbered 1-99. Odd numbers (like 5 or 95) go north/south, and evens (like 10 or 82) go east/west. Auxiliary highways are numbered 100-999, and service the primary highway indicated by the rightmost two digits. Thus, I-405 services I-5, and I-290 services I-90.

   Note: 200 is not a valid auxiliary highway because 00 is not a valid primary highway number.

   Let the user pick a highway number. Given a valid highway number, indicate whether it runs north/-south or east/west. If it is an invalid highway number, indicate that it is an invalid highway number.
   For example,

```
Pick a highway number: 400
Invalid highway number
```

```
Pick a highway number: 305
highway 305 runs north/south
```

```
Pick a highway number: 694
highway 694 runs east/west
```

```
Pick a highway number: 35
highway 35 runs north/south
```

3. (3.3) The table below shows the maximum health of characters based on race and class for a new video game I am creating. Write a program that asks the user for the race and the class of their character, and then sets the *health_points* variable according to the table below.

   health_points = -1
   #Your code here.

	Race	
Class	Elf	Ogre
Warrior	150	200
Bard	75	100
Wizard	25	50

President Skroob
section 1

Branching quiz

1. (3.1) Write a program that prompts the user for a letter and checks whether the letter is a vowel or consonant. A vowel should output "*vowel*", and a consonant should output *consonant*. You may assume only lower case letters. Below is sample output.
   Hint: In the English language, a, e, i, o, and u are the vowels.

```
Enter a letter: a
vowel
```

```
Enter a letter: x
consonant
```

2. (3.2) Primary U.S. interstate highways are numbered 1-99. Odd numbers (like 5 or 95) go north/south, and evens (like 10 or 82) go east/west. Auxiliary highways are numbered 100-999, and service the primary highway indicated by the rightmost two digits. Thus, I-405 services I-5, and I-290 services I-90.

   Note: 200 is not a valid auxiliary highway because 00 is not a valid primary highway number.

   Let the user pick a highway number. Given a valid highway number, indicate whether it runs north/-south or east/west. If it is an invalid highway number, indicate that it is an invalid highway number. For example,

```
Pick a highway number: 400
Invalid highway number
```

```
Pick a highway number: 305
highway 305 runs north/south
```

```
Pick a highway number: 694
highway 694 runs east/west
```

```
Pick a highway number: 35
highway 35 runs north/south
```

3. (3.3) The table below shows the maximum health of characters based on race and class for a new video game I am creating. Write a program that asks the user for the race and the class of their character, and then sets the *health_points* variable according to the table below.

health_points = -1
#Your code here.

	Race	
Class	Elf	Ogre
Warrior	150	200
Bard	75	100
Wizard	25	50