1. Compare and contrast `while` and `for` loops.

2. Compare and contrast `while` loops and `if` statements.

3. What is the output of the following code? If there are any errors, correct them.

```python
num = 2
while num < 50:
    print(num, end=" ")
    num = num * 2
```

4. Rewrite the above code replacing the `while` loop with a `for` loop:
5. Rewrite the following program replacing the `for` loop with a `while` loop:

```python
s = "abcxyz"
for ch in s:
    print(ch, end=" ")
```

6. Rewrite the following program replacing the `for` loop with a `while` loop:

```python
L = ['a', 'b', 'c', 'd', 'e']
for i in range(len(L)):
    print(L[-(i+1)], end=" ")
```

7. Rewrite the following program replacing the `for` loop with a `while` loop:

```python
for i in range(5):
    print(2 * (i + 1), end=" ")
```
8. When should you use a `for` loop and when should you use a `while` loop?

9. Write a program that asks the user for a lowercase string and prints each letter up to but not including the first vowel. Do not use `find()` or string slicing.

10. Write a function called `significance` that takes an int, `n`, and uses a `while` loop to return a string containing digits of `n` rearranged from least to most significant. For example:

    `significance(4982)` returns `'2894'`
Recall the HW 04 Game Winner program. This program kept track of each player's scores during a game and computed the winner at the end. In that homework there was a maximum of three rounds, now that we know while loops we can write a version with unlimited rounds.

11. Write a pure function called `update_scores` that takes a list of scores and the number of the current round (an int). Your function then asks the user for each player's score in this round and returns a new list with the updated score values.

12. Write a function called `game_winner` that takes a list of scores and the maximum number of points needed to win (an int). Your function will return a the index of the player who has won the game or False if there is no winner yet.

13. Write a function called `play_game` that takes no parameters. This function will ask the user for the number of players and the maximum score needed to win. It will then use the functions you wrote above to continue playing the game until a player wins. When a player wins, your function will print the number of the winning player and their score, along with the score of all of the other players.