Computer Science

COMPSCI 101 2020 S1

Assignment 5

Due: 4:30pm, Friday 12th June 2020.
Worth: This assignment is marked out of 30 and is worth 3% of your final mark.

- In your program there must not be any variables used outside any of the functions.
- From the CompSci 101 Assignment website: https://www.cs.auckland.ac.nz/courses/compsci101s1c/assignments/, download the skeleton Python program and rename the SkeletonA5.py file to "YourUsernameA5.py", e.g., afer023A5.py.
- Your program must include a docstring at the top of the file containing your name, your username and a correct description of the program.

Submission

Submit your Python program, your own palette text file and your own pattern text file using the Assignment Dropbox: https://adb.auckland.ac.nz/.

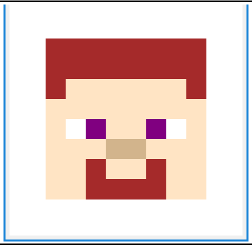
This assignment first reads a list of colours from a palette text file and creates a list of colour strings. For example, the `steve_palette.txt` contains the following lines:

```
1:brown
2:bisque
3:white
4:purple
5:tan
```

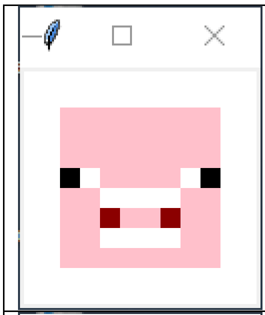
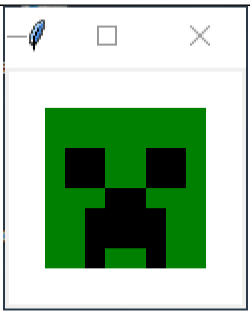
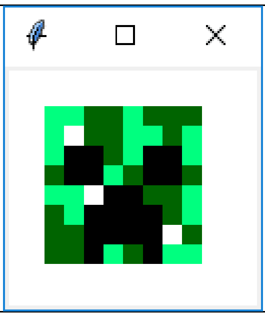
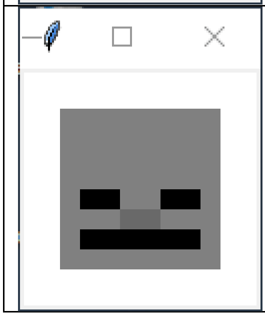
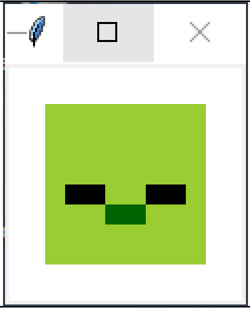
and returns a dictionary of colours:

```
{1: 'brown', 2: 'bisque', 3: 'white', 4: 'purple', 5: 'tan'}
```

Next, your program reads lines of digits from a pattern text file and draws a grid of coloured rectangles based on the information stored in the text file. Each row of the picture corresponds to a line in the file, and each rectangle along the row corresponds to a digit in the line. For example, if the colour codes that we read from the above palette file are `1: brown, 2: bisque, 3: white, 4: purple, 5: tan`, the following table shows the contents of the pattern file, "`steve.txt`" and the corresponding pixel art. The first row (11111111) represents 8 brown rectangles.

steve.txt	Pixel Art
11111111 11111111 12222221 22222222 23422432 22255222 22122122 22111122	

Here are some more examples:

	pig.txt 11111111 11111111 11111111 23111132 11333311 11455411 11333311 11111111		creeper1.txt 11111111 11111111 12211221 12211221 11122111 11222211 11222211 11211211		creeper2.txt 11221222 13221121 14421441 24412442 11344221 21444421 22444432 22412411
	skeleton.txt 11111111 11111111 11111111 11111111 12211221 11133111 12222221 11111111		zombie.txt 11111111 11111111 11111111 11111111 12211221 11133111 11111111 11111111		

Steps:

1. Add your username to the title bar of the window **(2 marks)**
Currently the title bar of the program window displays "A5 by". Add your username to the title bar of the program, i.e., the title bar should display the string, "A5 by yourUsername", e.g., A5 by afer023.

2. Add a docstring at the top of the file containing your name, your username and a correct description of the program. **(2 marks)**

3. Complete the `split_digits(line)` function **(2 marks)**
This function takes a String of digits as a parameter and returns a list of integer elements. For example, the output produced by the following call to the function is shown below:

```
print("3.", split_digits('11155111'))
3. [1, 1, 1, 5, 5, 1, 1, 1]
```

4. Complete the `process_file(filename)` function **(3 marks)**
This function takes a filename as a parameter and returns a list of strings read from the input file. For example, the output produced by the following call to the function is shown below:

```
lines = process_file("steve.txt")
print("4.", lines)
4. ['11111111', '11111111', '12222221', '22222222', '23422432', '22255222',
'22122122', '22111122']
```

5. Complete the `create_colours_dict(lines)` function **(3 marks)**
This function takes a list of strings as a parameter and returns a colours dictionary from the parameter string. For example, the output produced by the following call to the function is shown below:

```
lines = ['1:brown', '2:bisque', '3:white', '4:purple', '5:tan']
colours_dictionary = create_colours_dict(lines)
print("5.", colours_dictionary)
5. {1: 'brown', 2: 'bisque', 3: 'white', 4: 'purple', 5: 'tan'}
```

6. Complete the `create_pattern_list(lines)` function **(5 marks)**

This function takes a list of strings as a parameter. Each element of the list is a list of digits. The list of digits is produced by calling the `split_digits()` function above. For example, the output produced by the following call to the function is shown below:

```
lines = ['11111111', '11111111', '12222221', '22222222', '23422432', '22255222',
'22122122', '22111122']
pattern_list = create_pattern_list(lines)
print("6.", pattern_list)
6. [[1, 1, 1, 1, 1, 1, 1, 1], [1, 1, 1, 1, 1, 1, 1, 1], [1, 2, 2, 2, 2, 2, 2, 1],
[2, 2, 2, 2, 2, 2, 2, 2], [2, 3, 4, 2, 2, 4, 3, 2], [2, 2, 2, 5, 5, 2, 2, 2], [2,
2, 1, 2, 2, 1, 2, 2], [2, 2, 1, 1, 1, 1, 2, 2]]
```

7. Complete the `draw_pattern(a_canvas, colours_dictionary, pattern_list, size, left, top)` function **(5 marks)**
This function is passed **SIX** parameters: the Canvas object, a colours dictionary, a list of string elements, and followed by three integer parameters. The function draws a grid of coloured rectangles inside the canvas area. The left-top position of the grid of coloured rectangles is given by the last two parameter values and the size of each rectangle is given by the size parameter. Once you have completed this function you should see rows of coloured rectangles in the canvas area.

8. Complete the `main` function **(3 marks)**
At the top of the `main` function, add code to prompt the user to enter the palette filename and the pattern filename as in the example below:

```
Enter a palette filename: steve_palette.txt
Enter a pattern filename: steve.txt
```

The `main()` function should then call the `process_file()` function to return a list of strings and then call the `create_colours_dict()` function to return a colours dictionary. Then, the `main()` function should then call the `process_file()` function again to return a list of strings and then call the `create_pattern_list()` function to create the corresponding pattern list. Use the colours dictionary and the pattern list to draw a specific pattern in the canvas area.

9. Create your own palette and pattern text files **(5 marks)**
Your pattern should show great preparation, creativity or effort. Submit your text files to the Assignment dropbox. Some examples are given as below: