CS5001 Project 3 Report

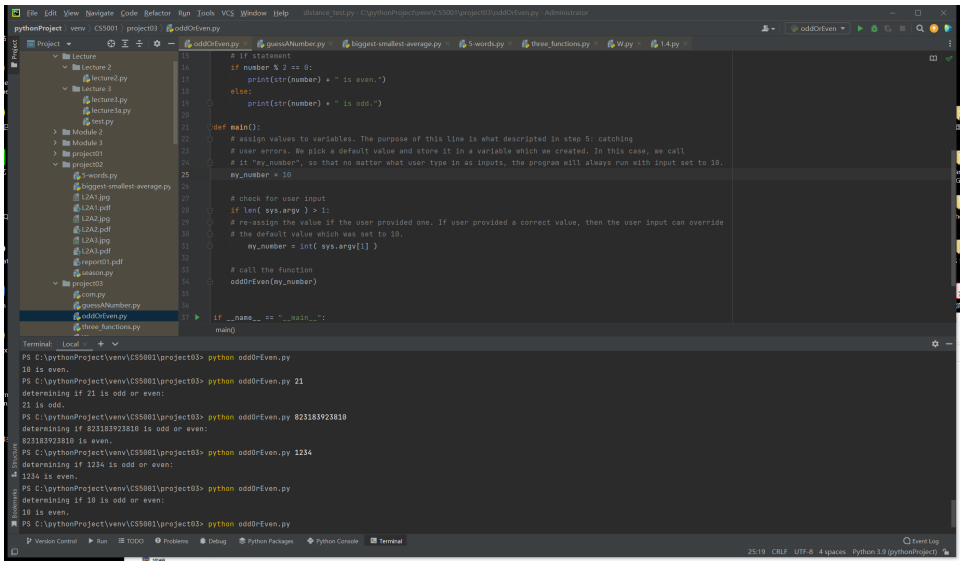
Shang Xiao

February 4, 2022

1 Problem Description

1.1 The first half of project 3 pushes us further by creating a lab environment that teaches us how to use command-line arguments in Terminal. One of the significant point in the walk-through tasks is to show us that users do not always do what they should in some cases. If we want to become better programmers and write better programs, we should assume our users are not smart, and we should always try to make our program as user-friendly as possible. This means even users do not give us what we want; we can still complete their requests by thinking ahead.

In this module, we learned a strategy to solve this problem by picking a default value and creating a variable to store that default value so that our program will always run even if users do not give us what we want. Then, if users give what we want, that valid input will replace the default value to return a favourable result. The figure below shows my implementation of this idea in Terminal.



1.2 The second half of project 3 asks us to develop a program that randomly picks a card from a typical 52 card deck. Since the deck is standard, we need to randomly pick one of the four suits and one of the thirteen values of each suit and print out that random card. Then we need to add a command-line asking users for their input, for the colour red or black. In the end, the program should be able to define three things:

What the randomly chosen card was, by the program.

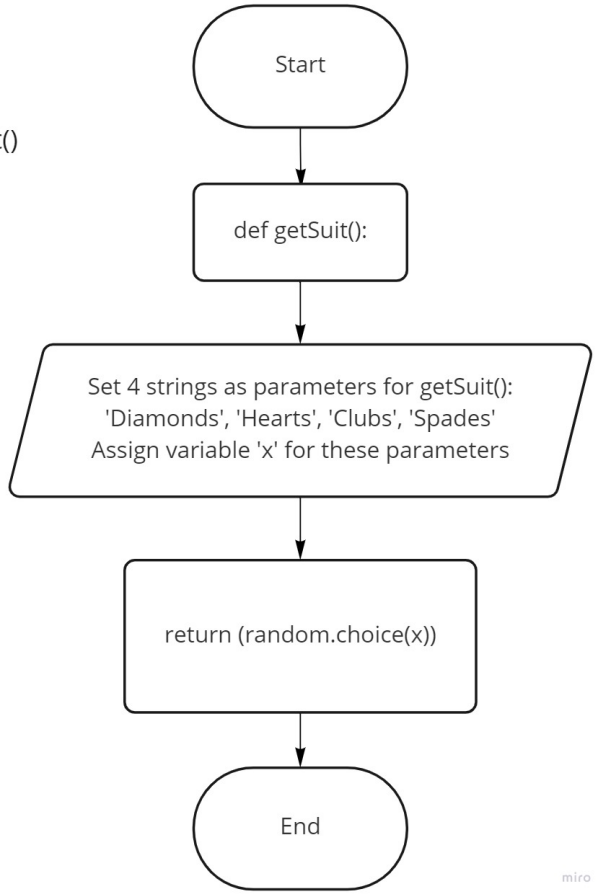
Which colour did the user pick, red or black?

If the randomly chosen card matches with the colour picked by the user.

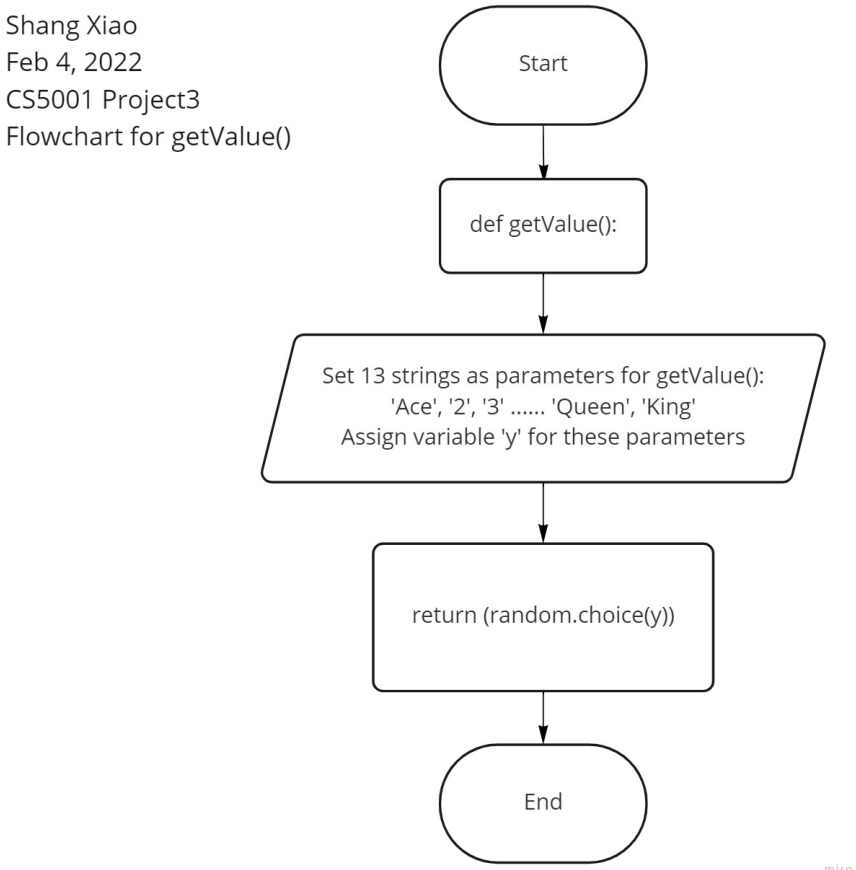
2 Activity Diagrams

2.1 The first function getSuit() pick a random suit for a card from Diamonds, Hearts, Clubs, and Spades. I used function 'random.choice' from the random module to accomplish this. The implementation was very straightforward as there were only four strings to consider, and the program successfully picked a random suit for one card. Below is my implementation flowchart for this function.

Shang Xiao
Feb 4, 2022
CS5001 Project3
Flowchart for getSuit()

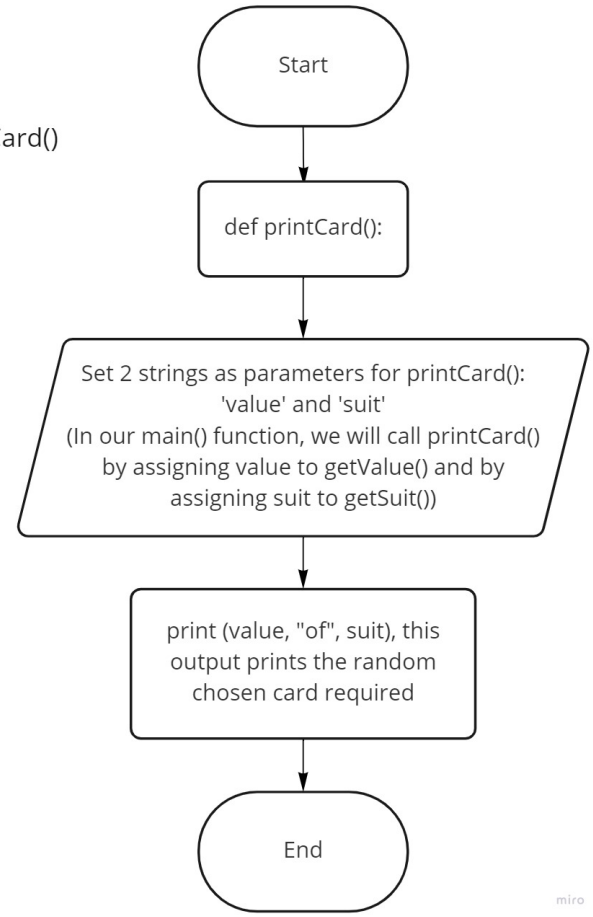


2.2 The second function getValue() pick a random value for a card from Ace to King. I also used function 'random.choice' from the random module to accomplish this. Similar to getSuit(), the implementation of this function was simple, and the program was able to pick a random value for one card successfully. Below is my implementation flowchart for this function.

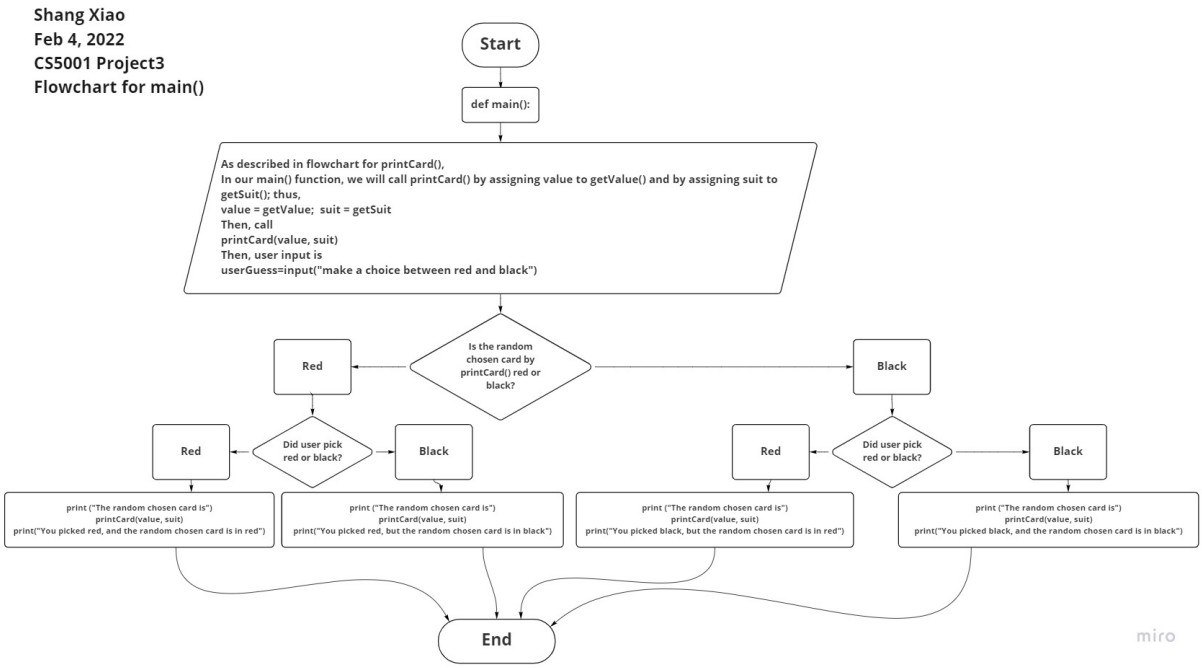


2.3 The third function printCard(value, suit) is relatively simple. It does one thing: combine the output result from getSuit() and getValue() into two parameters and print out these two parameters by using the print() function. Below is my implementation flowchart for this function, and the program was able to print out the randomly chosen card successfully.

Shang Xiao
Feb 4, 2022
CS5001 Project3
Flowchart for printCard()



2.4 The main() function integrates all three functions above and then asks the user to pick a colour between red and black. I used a set of 'if' and 'else' statements to connect the user input with the randomly chosen card from printCard(). At first, I could not figure out how to connect printCard() with user input and final output until I realized that I could use 'if' statement within another 'if' statement. After implementing the flowchart below, the main() function distinguished the user input and connected the randomly chosen card from printCard() to its corresponding scenarios.



3 Reflection

3.1 By the end of this project, I was able to understand the idea of 'override' from a programmer's point of view. The key idea behind this is to set a default value first to ensure the primary functionality of functions, then expand on that primary functionality to accomplish user requests.

I further developed my logical thinking in the deck card project by enhancing my function writing skills. The project enables me to define three elements of a function more quickly. In addition, I can now write functions more proficiently and structure my code more clearly by building organized flowcharts.

I am looking forward to explore more advanced function structures in my future studies.

4 Acknowledgements

4.1 Website consulted (lecture notes and documentations can be found in these links):

https://www.geeksforgeeks.org/
https://northeastern.instructure.com/courses/102943
https://greenteapress.com/thinkpython2/html/index.html

4.2 Website used for flowchart: https://miro.com/app/dashboard/