

DICE GAME

PROJECT NUMBER: 1



LOVELY
PROFESSIONAL
UNIVERSITY

Transforming Education Transforming India

S.NO	NAME	ADMISSION NUMBER	SIGN.
1	VARSHITH	12205084	
2	SAHITH REDDY	12217178	
3	ABHILASH	12213877	

SECTION: K22CS
SUBJECT: INT108

**SUBMITTED TO NISHANT AGNIHOTRI, ASSISTANT
PROFESSOR**

**LOVELY PROFESSIONAL UNIVERSITY, JALANDHAR
PUNJAB**

INDEX

1.Acknowledgement

2.*Introduction*

3.Methodology

4.Flow Chart

5.Code Listing

6.Conclusion

1. ACKNOWLEDGEMENT

I would like to acknowledge all those without whom this project would not have been successful. Firstly, I would wish to thank our Computer Science teacher Mr. Nishant Agnihotri who guided me throughout the project and gave his/her immense support. He made us understand how to successfully complete this project and without him, the project would not have been complete.

This project has been a source to learn and bring our theoretical knowledge to the real-life world. So, I would really acknowledge his help and guidance for this project.

Once again, thanks to everyone for making this project successful.

2. INTRODUCTION

PYTHON SIMPLE DICE GAME

Here's a simple program in Python that demonstrates some important fundamental concepts in computer programming. The program simulates two players taking it in turns to roll a 6-sided dice and keeps track of the scores. It makes use of the three basic control structures of sequence, selection and iteration.

SEQUENCE

This involves thinking about the order in which instructions need to be executed to produce the desired result.

SELECTION

This involves choosing (selecting) different paths through a program based on conditions specified by the programmer.

ITERATION

One of the most powerful aspects of computers is their ability to perform repetition. Iteration is a kind of repetition which often involves loops such as for loops and while loops.

As well as these 3 basic control structures, This simple program covers some other important fundamental programming skills.

- Creating variables, using descriptive names
- Working with strings and integers
- Outputting information to the screen
- Updating the value of a variable

The Python random module is used to simulate rolling a 6-sided dice using `random.randint(1, 6)`.

The basic idea is that you roll dice of two players and the results are given. If player 1 gets more points than player 2 than player one wins. Same as the player 2 gets more points than player 1, player 2 wins. If the points are same than it is draw. At last, the results are given as how many times the player won.

3. METHODOLOGY

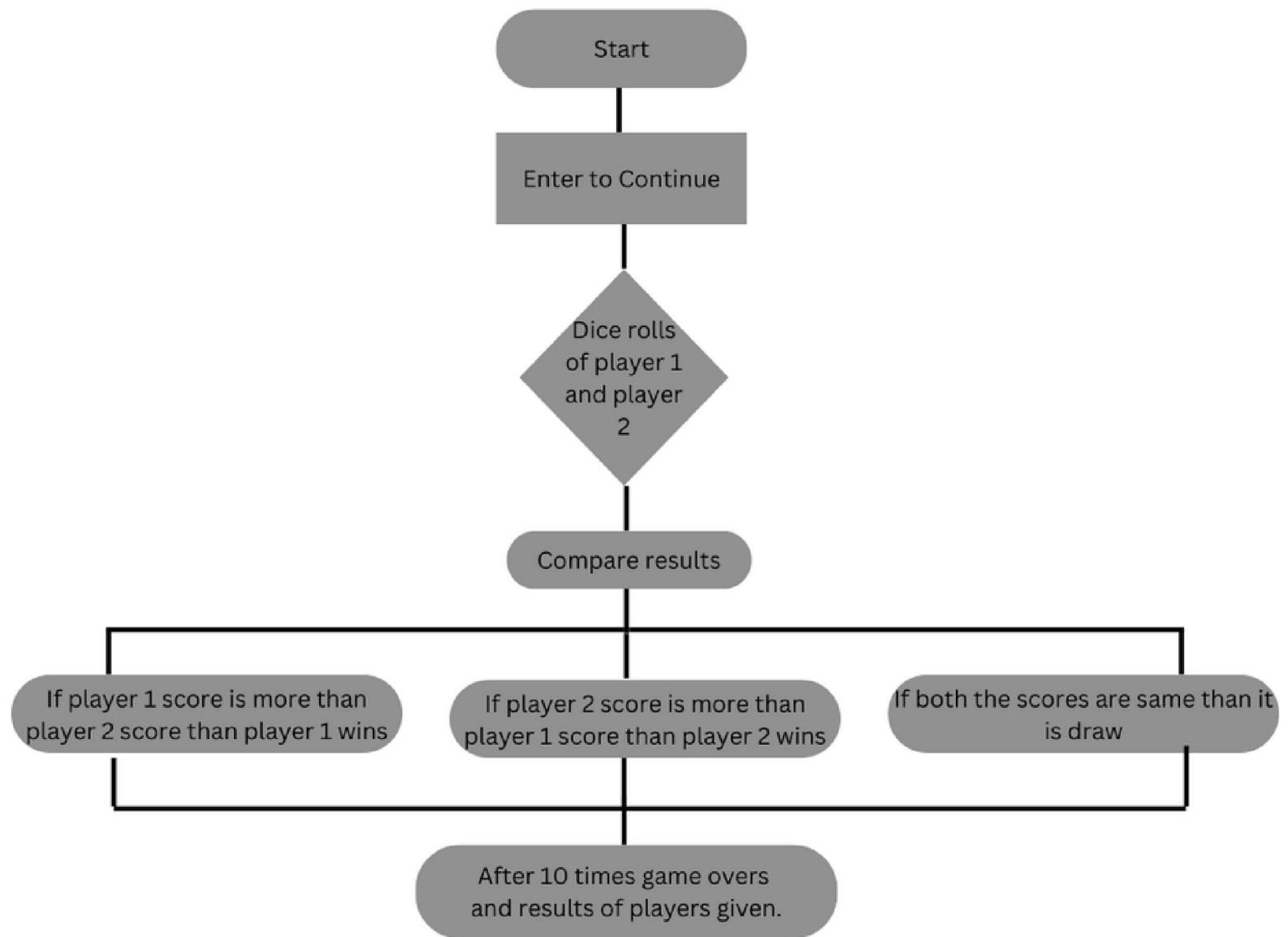
This has 3 basic control structures, this simple program covers some other important fundamental programming skills.

- Creating variables, using descriptive names
- Working with strings and integers
- Outputting information to the screen
- Updating the value of a variable
- Creating random values using Python's random module
- Python comparison operators

The Python random module is used to simulate rolling a 6-sided dice using `random.randint(1, 6)`. According to the docs, this function Gets a random integer in the range [a, b] including both end points.

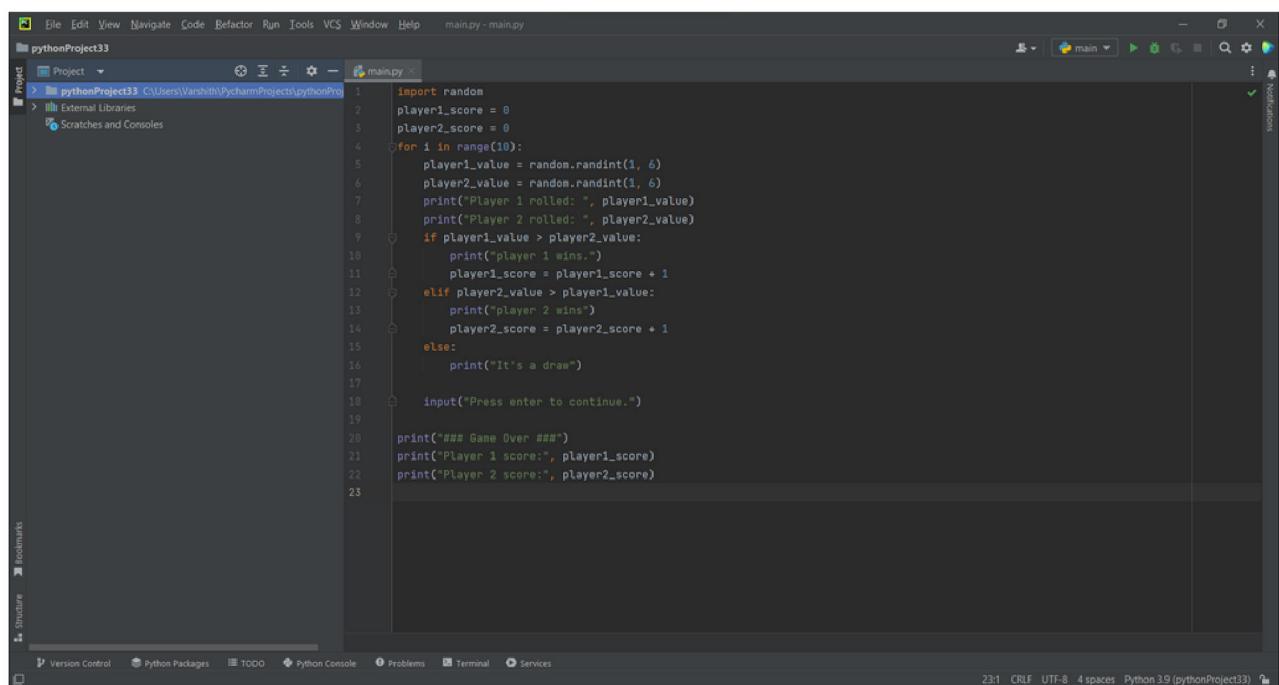
TOOL: PYCHARM

4.FLOW CHART



5.CODE LISTING

1. We import random because we need to create random numbers to simulate the dice roll
2. Next we put players initial scores as 0
3. Now we use 'for' loop to repeat everything in 10 times after 10 times the game ends.



The screenshot shows the PyCharm IDE interface with the main.py file open. The code implements a dice game where two players roll a six-sided die 10 times each. The player with the highest roll in a single turn wins that round. If both players roll the same value, it's a draw. The total score for each player is calculated at the end of the 10 rounds. The PyCharm interface includes toolbars, a project tree, and various status indicators at the bottom.

```
import random
player1_score = 0
player2_score = 0
for i in range(10):
    player1_value = random.randint(1, 6)
    player2_value = random.randint(1, 6)
    print("Player 1 rolled: ", player1_value)
    print("Player 2 rolled: ", player2_value)
    if player1_value > player2_value:
        print("Player 1 wins.")
        player1_score = player1_score + 1
    elif player2_value > player1_value:
        print("Player 2 wins")
        player2_score = player2_score + 1
    else:
        print("It's a draw")

    input("Press enter to continue.")

print("## Game Over ##")
print("Player 1 score:", player1_score)
print("Player 2 score:", player2_score)
```

Fig 5.1: Code belongs to the dice game python project

4. Under 'for' loop we use random.randint to get random values for the players from 1 to 6

5. Now we display the values of both players

6. Now we use 'if', 'elif' and 'else', print the output as if player 1 gets more than player 2 then prints player 1 wins

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help main.py - main.py
pythonProject33
Project External Libraries Scratches and Consoles
main.py
11     player1_score = player1_score + 1
12 elif player2_value > player1_value:
13     print("Player 2 wins")
14     player2_score = player2_score + 1
15 else:
16     print("It's a draw")
17
18     input("Press enter to continue.")
19
20 print("### Game Over ###")
21 print("Player 1 score:", player1_score)
22 print("Player 2 score:", player2_score)
23
Run main
C:\Users\Varshith\PycharmProjects\pythonProject33\venv\Scripts\python.exe C:\Users\Varshith\PycharmProjects\pythonProject33\main.py
Player 1 rolled: 4
Player 2 rolled: 4
It's a draw
Press enter to continue.
Run: main
Version Control Run Python Packages TODO Python Console Problems Terminal Services
525 CRLF UTF-8 4 spaces Python 3.9 (pythonProject33)
```

Fig 5.2: Working example of code

CONCLUSION

This very simple game is just another way to try and understand computer science and programming. Like the Hello World examples, a simple dice game is one of the many ways to start your journey, and I hope you've learned a bit from this one as well.

GIT HUB LINK:

GITHUB.COM/VARSHITH9030/DICE-GAME-PYTHON-PROJECT