

# Python Project: Racquetball

## Concept

In this project you will create a python program that runs through several games of racquetball. This project challenges your knowledge of mathematical abilities, variables, return statements, and value comparisons.

Overall, the program should take about 70 lines of code to complete, and will require several functions to be written. The user will determine the probabilities for both players, and the total number of games to be played.

## Instructions

1. Start by implementing `getInputs()`, and check to make sure you are returning all three values (A, B, and N) as integers, not strings.
2. Then, work on `simOneGame()`, passing the values given to you by the user as input.
  - a. Your while loop should look something like this:  

```
while not gameOver(scoreA, scoreB):
```
  - b. Remember to return the scores for players A and B after you've completed the game
3. Make sure `gameOver()` is functioning correctly, which should return True when either player has scored 15 points, and False otherwise.
4. Finally, work on `simNGames()`, which should loop through N number of games, where N is the number of games to play given to you by the user.
  - a. Like before, remember to return the total wins for both players A and B

# Code

```
import random

def main():
    printIntro()
    probA, probB, n = getInputs()
    winsA, winsB = simNGames(n, probA, probB)
    printSummary(winsA, winsB)

def printIntro():
    print("This program simulates a game of racquetball between
two")
    print('players called "A" and "B". The abilities of each player
is')
    print("the player wins the point when serving. Player A
always")
    print("has the first serve.")

def getInputs():
    # Returns the three simulation parameters

    """
    * Declare three variables: a, b, and n

    * a is the probability that player A wins on a serve

    * b is the probability that player B wins on a serve

    * n is the number of games you will simulate
    """

    """
    * Return a, b, and n
    """

def simNGames(n, probA, probB):
    # Simulates n games of racquetball between players whose
    # abilities are represented by the probability of winning a
serve
    # Returns number of wins for A and B

    """
    * Declare two variables: winsA, and winsB
```

```

    * winsA is the number of wins player A has,
    which should start at 0

    * winsB is the number of wins player B has,
    which should start at 0
    """

    """
    * Create a for loop in range of the number of games played

    * call simOneGame and save the scores returned as scoreA and
scoreB

    * if player A has a higher score than player B, increment winsA
by 1

    * otherwise, increment winsB by 1

    * Then, once all games have been played, return the total wins
for both
    players A and B
    """

def simOneGame(probA, probB):
    # Simulates n games of racquetball between players whose
    # abilities are represented by the probability of winning a
serve
    # Returns number of wins for A and B

    """
    * Declare three variables: serving, scoreA, and scoreB

    * serving should be initialized as "A"

    * scoreA and scoreB should be initialized as 0

    * Create a while loop that checks if gameOver() returns false

    * If player A is serving,

        * Generate a random value from 0 to 100

        * If that random value is less than probA, give player A a
point

```

```

        * Otherwise, switch serving to equal "B"

    * Otherwise, player B is serving

        * Generate a random value from 0 to 100

        * If that random value is less than probB, give player B a
point

        * Otherwise, switch serving to equal "A"

    * Once the game is over, return the scores for player A and B
    """

def gameOver(a, b):
    # a and b represent scores for racquetball game
    # Returns True if either player scores 15 points, False
otherwise
    """

def printSummary(winsA, winsB):
    # Prints a summary of wins for each player
    n = winsA + winsB
    print("\nGames simulated:", n)
    print("Wins for A: {0} ({1:0.1%})".format(winsA, winsA/n))
    print("Wins for B: {0} ({1:0.1%})".format(winsB, winsB/n))

if __name__ == "__main__":
    main()

```