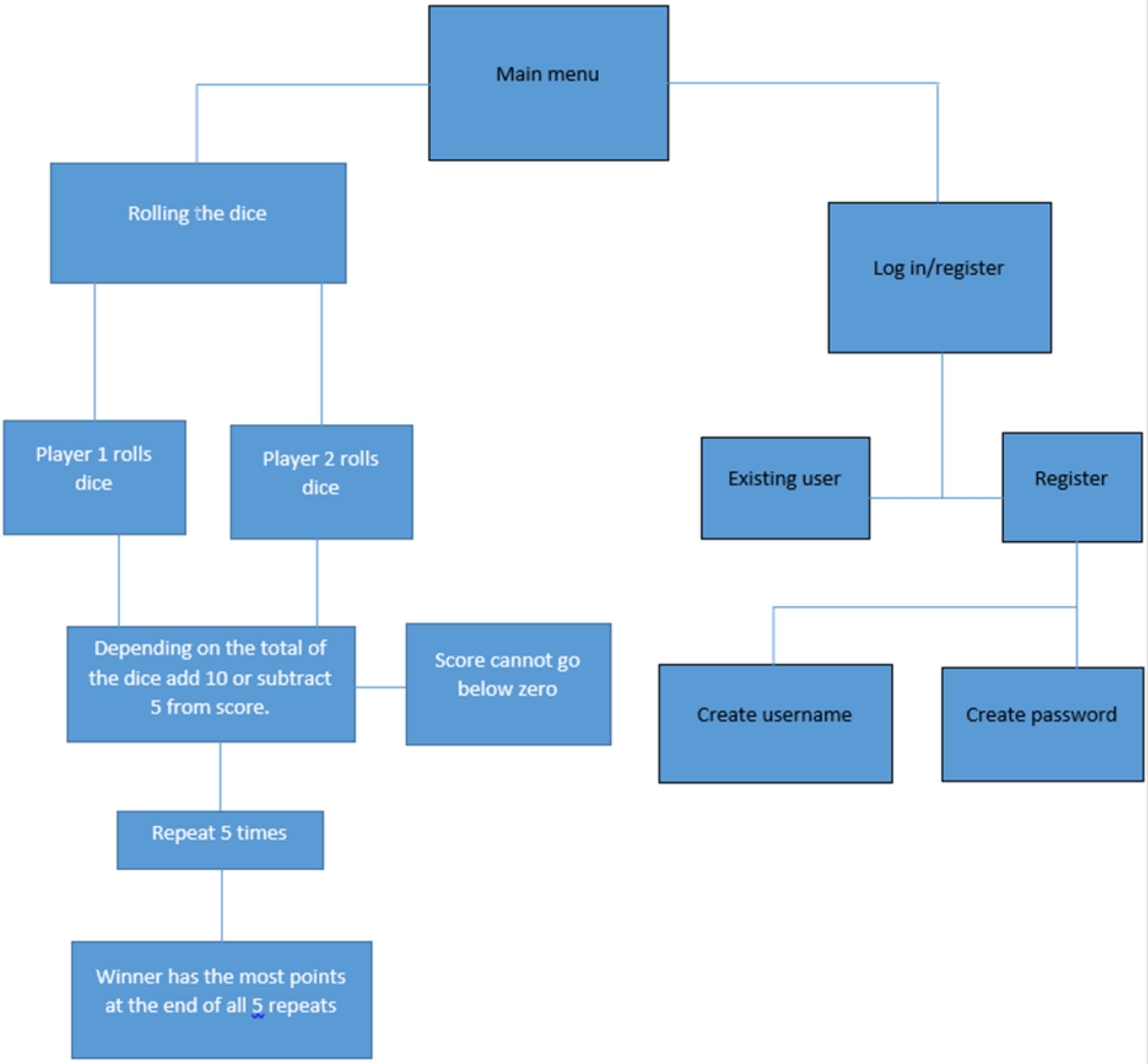
Programming project

Analysis:

Introduction: The game we are programming is a two-player dice game. The players can only get points if total of the two di is even. If the total is odd, then points are subtracted from the total. The game will end after five rounds and whoever has the highest score. To make sure it all works I will use a testing table.

Abstraction - removing all the unnecessary details and to only focus on the relevant details in problem solving.

Decomposition - making a complex situation easier by breaking it down into smaller, easier parts

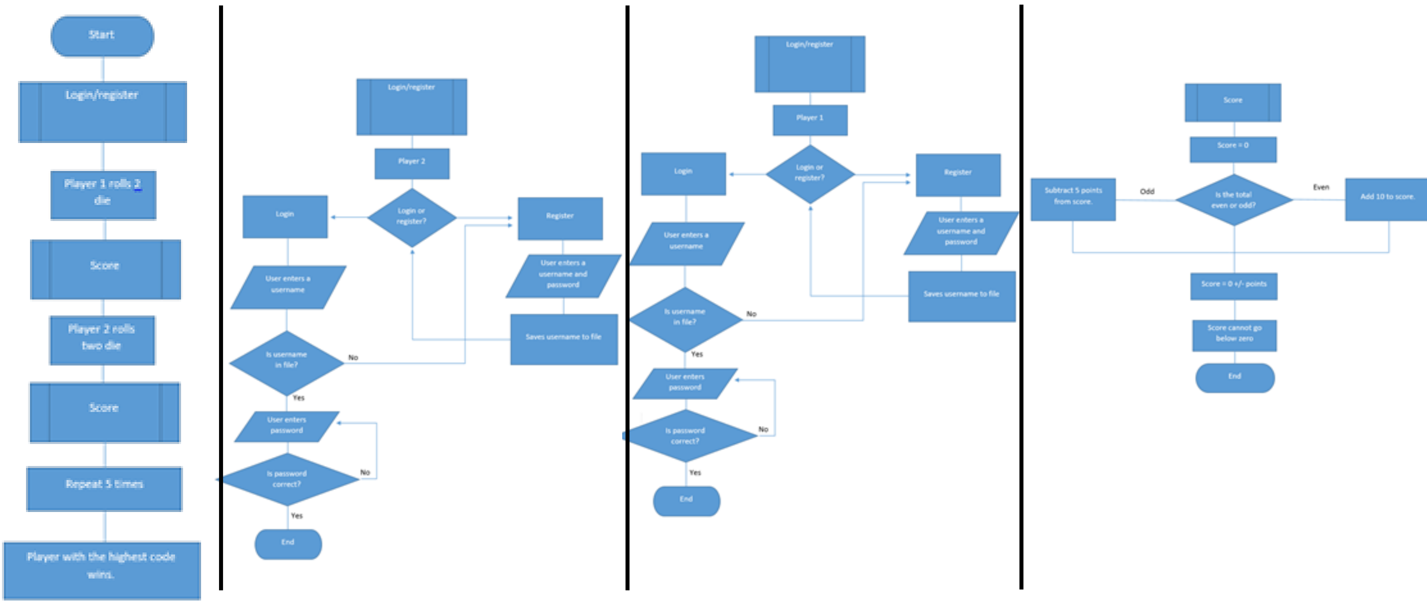

Main menu
Rolling the dice
Log in/register
Player 1 rolls dice
Player 2 rolls dice
Existing user
Register
Depending on the total of the dice add 10 or subtract 5 from score.
Score cannot go below zero
Create username
Create password
Repeat 5 times
Winner has the most points at the end of all 5 repeats

Design:

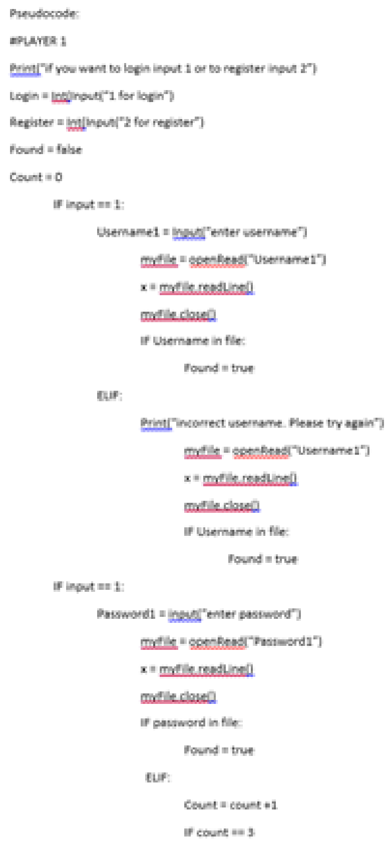
	Success criteria	How I met them
1	Can only start the game once two people have logged in.	I created a menu, login and register function that is used before the game function.
2	Store the high scores	I created code that stores the highest score in the game to the person who rolled the dice.
3	Roll two dice	I created 2 dice variables that a rolled each round
4	Adds and deducts points correctly	I created code that adds 10 point to the score when an even number is rolled and subtracts 5 when an odd number is rolled.
5	2 players have to register	I created code where once the first player is registered or logged in, the second player also has to either register or login.
6	Print player…. won the game	I added all the score together from each round and whoever had the highest score won and it printed their name.
7	Print player…. lost the game	The person with the least score lost the game and it printed which player.
8	Print what numbers were rolled	After each roll the programme prints what numbers where printed.
9	Print the points of each player after each roll	I met this after each round I printed whether the player received -5 or +10 points.
10	Players score cannot go below zero	I met this by creating code that does not allow score to go below 0
11	If even 10 points are added to players score	I met this code after each round when the score is even 10 points are added.
12	If odd 5 points are deducted from players score.	I met this code after each round when the score is even 5 points are subtracted.
13	They play 5 rounds	I met this by creating code before the rounds started that if rounds goes above 5 the round stops and player 2 has their turn.
14	If score is equal at the end they each roll a dice and whoever gets the highest score wins	I met this at the end of the code by seeing who has the highest score after another roll.
15	A main menu appears for the users	At the star of the code, a main menu is the first thing, which allows the user to either log in or register.
16	Usernames are stored in external files	After the user inputs their username, an external file is opened and it is written to it.
17	Top scores are saved into external files	After the game, the high scores are assigned to the username in the external file.

18	Allows user to enter username and password	When login in, the user is required to input their username and password.
19	Passwords are stored in external files	When the password is inputted, it is saved to the external file and saved to the username.
20	Player with the highest score wins	I met this by creating code that he highest score at the end of the game wins.

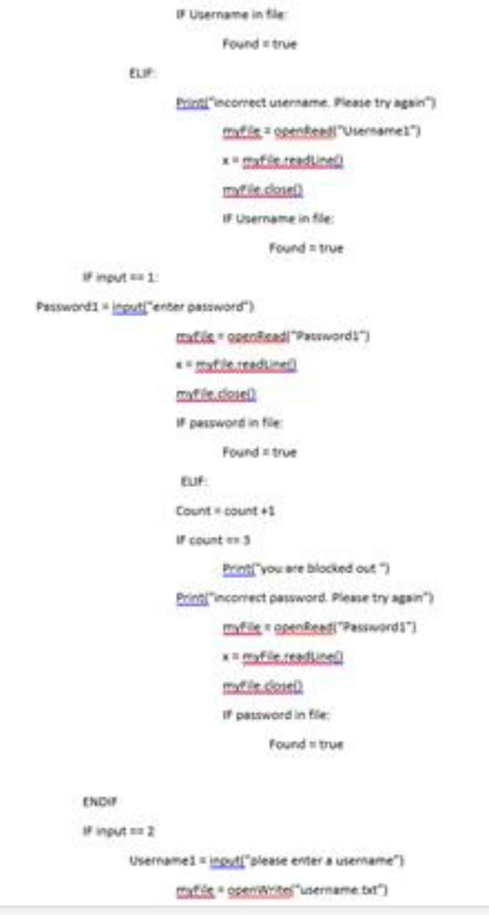
Flow chart:



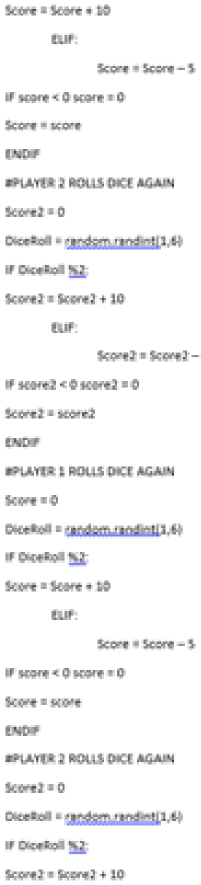
Pseudocode:



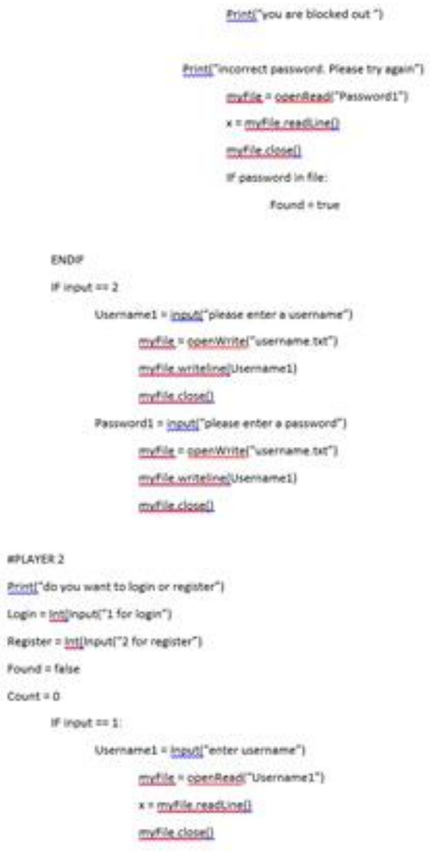
```
Pseudocode:
#PLAYER 1
Print("if you want to login input 1 or to register input 2")
Login = int(input("1 for login")
Register = int(input("2 for register")
Found = false
Count = 0
    IF input == 1:
        Username1 = input("enter username")
            myfile = openRead("Username1")
            x = myFile.readLine()
            myfile.close()
            IF Username in file:
                Found = true
        ELIF:
            Print("incorrect username. Please try again")
                myfile = openRead("Username1")
                x = myFile.readLine()
                myfile.close()
                IF Username in file:
                    Found = true
    IF input == 1:
        Password1 = input("enter password")
            myfile = openRead("Password1")
            x = myFile.readLine()
            myfile.close()
            IF password in file:
                Found = true
             ELIF:
                Count = count +1
                IF count == 3
```



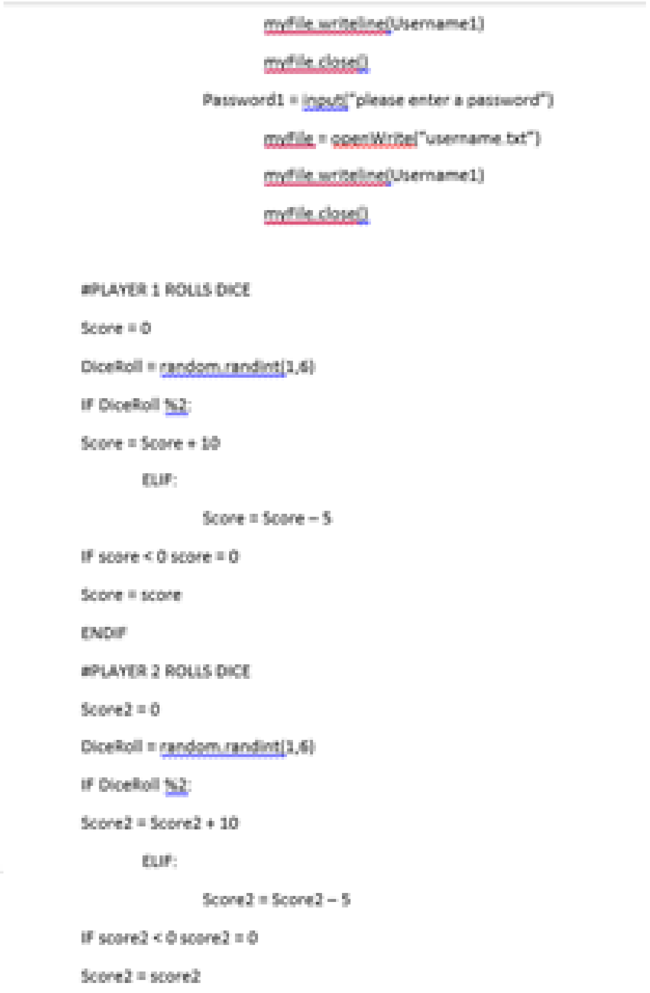
```
            IF Username in file:
                Found = true
        ELIF:
            Print("incorrect username. Please try again")
                myFile = openRead("Username1")
                x = myFile.readLine()
                myFile.close()
                IF Username in file:
                    Found = true
    IF input == 1:
Password1 = input("enter password")
            myFile = openRead("Password1")
            x = myFile.readLine()
            myFile.close()
            IF password in file:
                Found = true
             ELIF:
            Count = count +1
            IF count == 3
                Print("you are blocked out ")
            Print("incorrect password. Please try again")
                myFile = openRead("Password1")
                x = myFile.readLine()
                myFile.close()
                IF password in file:
                    Found = true

    ENDIF
    IF input == 2
        Username1 = input("please enter a username")
            myFile = openWrite("username.txt")
```



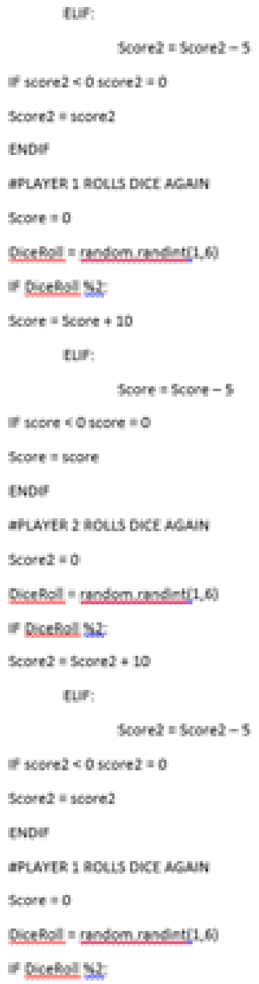
```
Score = Score + 10
    ELIF:
        Score = Score – 5
IF score < 0 score = 0
Score = score
ENDIF
#PLAYER 2 ROLLS DICE AGAIN
Score2 = 0
DiceRoll = random.randint(1,6)
IF DiceRoll %2:
Score2 = Score2 + 10
    ELIF:
        Score2 = Score2 – 5
IF score2 < 0 score2 = 0
Score2 = score2
ENDIF
#PLAYER 1 ROLLS DICE AGAIN
Score = 0
DiceRoll = random.randint(1,6)
IF DiceRoll %2:
Score = Score + 10
    ELIF:
        Score = Score – 5
IF score < 0 score = 0
Score = score
ENDIF
#PLAYER 2 ROLLS DICE AGAIN
Score2 = 0
DiceRoll = random.randint(1,6)
IF DiceRoll %2:
Score2 = Score2 + 10
```



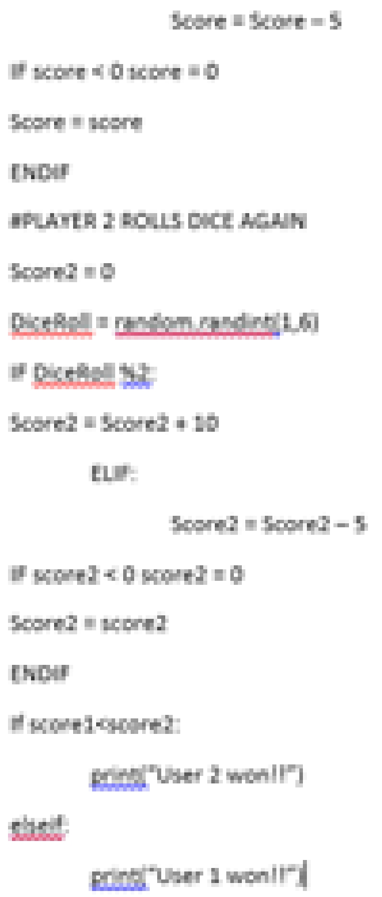
```
                    Print("you are blocked out ")

                Print("incorrect password. Please try again")
                    myfile = openRead("Pasword1")
                    x = myFile.readLine()
                    myfile.close()
                    IF password in file:
                        Found = true

    ENDIF
    IF input == 2
        Username1 = input("please enter a username")
            myfile = openWrite("username.txt")
            myfile.writeline(Username1)
            myfile.close()
        Password1 = input("please enter a password")
            myfile = openWrite("username.txt")
            myfile.writeline(Username1)
            myfile.close()

#PLAYER 2
Print("do you want to login or register")
Login = int(input("1 for login")
Register = int(Input("2 for register")
Found = false
Count = 0
    IF input == 1:
        Username1 = input("enter username")
            myfile = openRead("Username1")
            x = myfile.readLine()
            myfile.close()
```



```
                myfile.writeline(Username1)
                myfile.close()
            Password1 = input("please enter a password")
                myfile = openWrite("username.txt")
                myfile.writeline(Username1)
                myfile.close()

#PLAYER 1 ROLLS DICE
Score = 0
DiceRoll = random.randint(1,6)
IF DiceRoll %2:
Score = Score + 10
    ELIF:
        Score = Score – 5
IF score < 0 score = 0
Score = score
ENDIF
#PLAYER 2 ROLLS DICE
Score2 = 0
DiceRoll = random.randint(1,6)
IF DiceRoll %2:
Score2 = Score2 + 10
    ELIF:
        Score2 = Score2 – 5
IF score2 < 0 score2 = 0
Score2 = score2
```



```
    ELIF:
        Score2 = Score2 – 5
IF score2 < 0 score2 = 0
Score2 = score2
ENDIF
#PLAYER 1 ROLLS DICE AGAIN
Score = 0
DiceRoll = random.randint(1,6)
IF DiceRoll %2:
Score = Score + 10
    ELIF:
        Score = Score – 5
IF score < 0 score = 0
Score = score
ENDIF
#PLAYER 2 ROLLS DICE AGAIN
Score2 = 0
DiceRoll = random.randint(1,6)
IF DiceRoll %2:
Score2 = Score2 + 10
    ELIF:
        Score2 = Score2 – 5
IF score2 < 0 score2 = 0
Score2 = score2
ENDIF
#PLAYER 1 ROLLS DICE AGAIN
Score = 0
DiceRoll = random.randint(1,6)
IF DiceRoll %2:
```



```
        Score = Score – 5
IF score < 0 score = 0
Score = score
ENDIF
#PLAYER 2 ROLLS DICE AGAIN
Score2 = 0
DiceRoll = random.randint(1,6)
IF DiceRoll %2:
Score2 = Score2 + 10
    ELIF:
        Score2 = Score2 – 5
IF score2 < 0 score2 = 0
Score2 = score2
ENDIF
If score1<score2:
    print("User 2 won!!")
elseif:
    print("User 1 won!!")
```

Test table:

Test number	Test type	Test description against user requirements	How the test is performed	Expected result	Actual result (pass/fail)	Further action needed
1	valid	Test whether the menu takes you to the login or register menu after the integer linked to a certain process is inputted.	The user inputs 1 or 2	Login or register menu	pass	no
2	Erroneous	Test whether the menu accepts letters	The user inputs a letter	Error, please re-input either 1 or 2	fail	yes
3	Valid	Is username in the file	The user enters a username then a print statement is used to see the contents of file	The name should be outputted	pass	no
4	Valid	The incorrect password entered	The user enters a password for the username, so the programme searches through the file for the password.	A print statement is outputted saying "incorrect password, please try again."	pass	no
5	Valid	To test what happens when the correct password is inputted	The user enters a password for the username, so the programme searches through the file for the password.	It logs the user in and asks player 2 to either login or register	pass	no

6	Boundary	To test whether the boundaries on the dice range from 1 to 6	The output of the result from the rolled dice	An integer between 1 and 6 is outputted	pass	no
7	Valid	If the total on the dice odd 5 points are subtracted from score.	The programme outputs an odd total.	Score = 0 because score cannot go below 0	pass	no
8	Valid	If the total on the dice even 10 points are added to score.	The programme outputs an even total.	Score = 10 as score was 0 at the start.	pass	no
9	Valid	Does the score go below 0	The programme runs the game and outputs the score.	Score does not go below 0.	pass	no

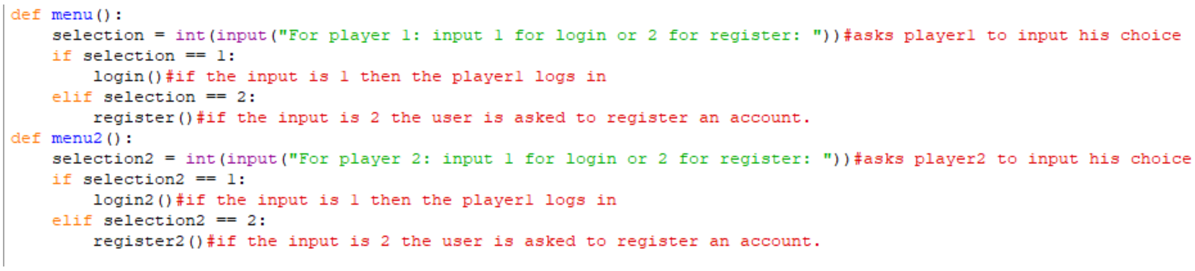
Variable name	Data type	Purpose
Login	string	Help user login
Register	String	Helps user register
Username1	String	Stores the 1st players username into username1 for further use later.
Myfile	String	It holds the code to search through the file for the username/password
X	String	Allows the programme to read the file.
Found	Integer	Will either be true or false depending on whether the username/password is found.
Username2	String	Stores the 2nd players username into username2 for further use later.
Password1	String	Stores the 1st players password for later use
Password2	String	Stores the 2nd players password for later use

Score	Integer	Never goes below 0 but either +10 or -5 depending whether the total of both dice rolls is either odd or even.
DiceRoll	Integer	Chooses a random number between 1 and 6
Score2	Inter	Never goes below 0 but either +10 or -5 depending whether the total of both dice rolls is either odd or even and stores it for the 2nd player.

Development:

Menu:

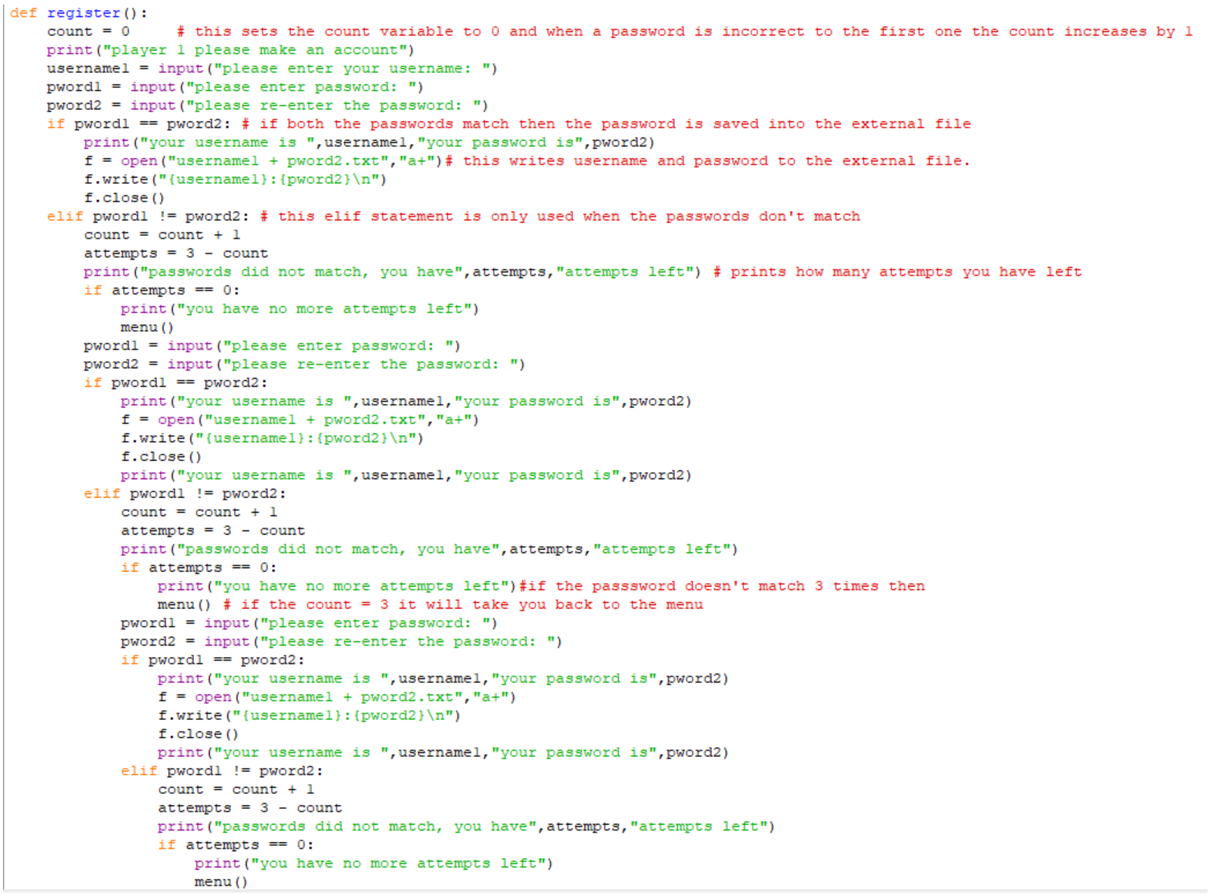
In these functions, it asks the first player to input whether they want to login or register and will the repeat for player two. This set of code set me up nicely to start the next function.



```
def menu():
    selection = int(input("For player 1: input 1 for login or 2 for register: "))#asks player1 to input his choice
    if selection == 1:
        login()#if the input is 1 then the player1 logs in
    elif selection == 2:
        register()#if the input is 2 the user is asked to register an account.
def menu2():
    selection2 = int(input("For player 2: input 1 for login or 2 for register: "))#asks player2 to input his choice
    if selection2 == 1:
        login2()#if the input is 1 then the player1 logs in
    elif selection2 == 2:
        register2()#if the input is 2 the user is asked to register an account.
```

Register:

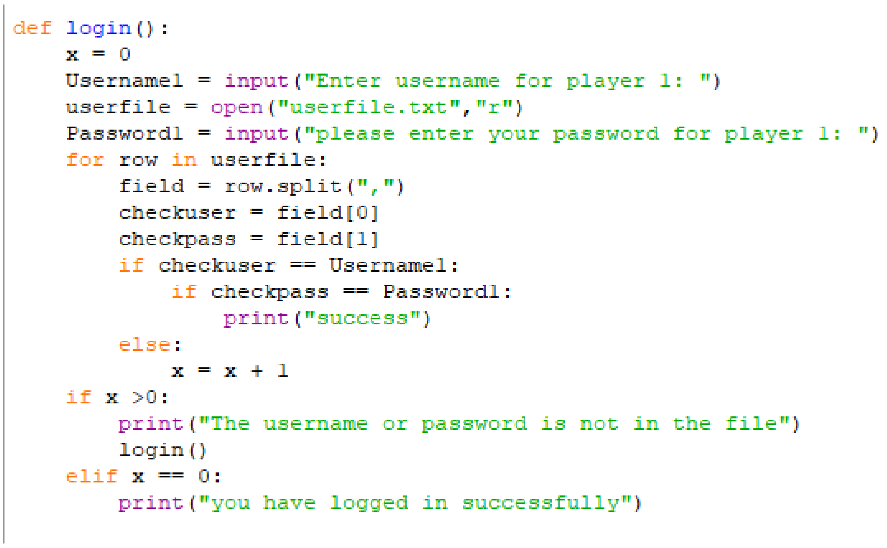
To start this function I asked the user to enter a desired username and password. Then I needed to save these details to a file but only if two passwords are the same, but I also had to produce more code for whether the passwords did not match each other. I had to repeat these 2 pieces if code 3 times because when the user fails to input the same password 3 times they are taken back to the main menu.



```
def register():
    count = 0     # this sets the count variable to 0 and when a password is incorrect to the first one the count increases by 1
    print("player 1 please make an account")
    username1 = input("please enter your username: ")
    pword1 = input("please enter password: ")
    pword2 = input("please re-enter the password: ")
    if pword1 == pword2: # if both the passwords match then the password is saved into the external file
        print("your username is ",username1,"your password is",pword2)
        f = open("username1 + pword2.txt","a+")# this writes username and password to the external file.
        f.write("{username1}:{pword2}\n")
        f.close()
    elif pword1 != pword2: # this elif statement is only used when the passwords don't match
        count = count + 1
        attempts = 3 - count
        print("passwords did not match, you have",attempts,"attempts left") # prints how many attempts you have left
        if attempts == 0:
            print("you have no more attempts left")
            menu()
        pword1 = input("please enter password: ")
        pword2 = input("please re-enter the password: ")
        if pword1 == pword2:
            print("your username is ",username1,"your password is",pword2)
            f = open("username1 + pword2.txt","a+")
            f.write("{username1}:{pword2}\n")
            f.close()
            print("your username is ",username1,"your password is",pword2)
        elif pword1 != pword2:
            count = count + 1
            attempts = 3 - count
            print("passwords did not match, you have",attempts,"attempts left")
            if attempts == 0:
                print("you have no more attempts left")#if the passsword doesn't match 3 times then
                menu() # if the count = 3 it will take you back to the menu
            pword1 = input("please enter password: ")
            pword2 = input("please re-enter the password: ")
            if pword1 == pword2:
                print("your username is ",username1,"your password is",pword2)
                f = open("username1 + pword2.txt","a+")
                f.write("{username1}:{pword2}\n")
                f.close()
                print("your username is ",username1,"your password is",pword2)
            elif pword1 != pword2:
                count = count + 1
                attempts = 3 - count
                print("passwords did not match, you have",attempts,"attempts left")
                if attempts == 0:
                    print("you have no more attempts left")
                    menu()
```

Login:

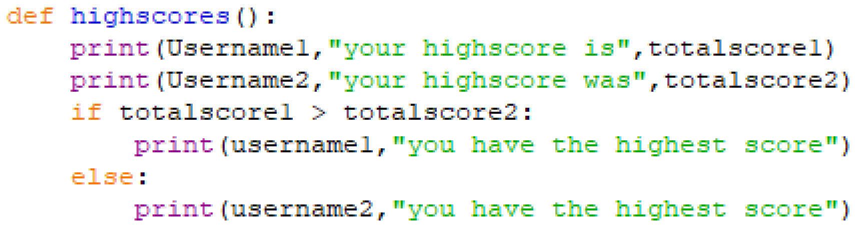
During the login function, I started by defining 'x' for later on in the code. I then asked the user to input their username and then reads the file for the username. It then asks the user to input their password. For the next part, I struggled a lot with it checking the username and password in the file; a lot of trial and error went into this stage of development. If the username is in the file then it prints success however if it is not correct, 'x' is incremented by 1 and the program outputs 'the user name is not in the file'



```
def login():
    x = 0
    Username1 = input("Enter username for player 1: ")
    userfile = open("userfile.txt","r")
    Password1 = input("please enter your password for player 1: ")
    for row in userfile:
        field = row.split(",")
        checkuser = field[0]
        checkpass = field[1]
        if checkuser == Username1:
            if checkpass == Password1:
                print("success")
        else:
            x = x + 1
    if x >0:
        print("The username or password is not in the file")
        login()
    elif x == 0:
        print("you have logged in successfully")
```

Highs cores:

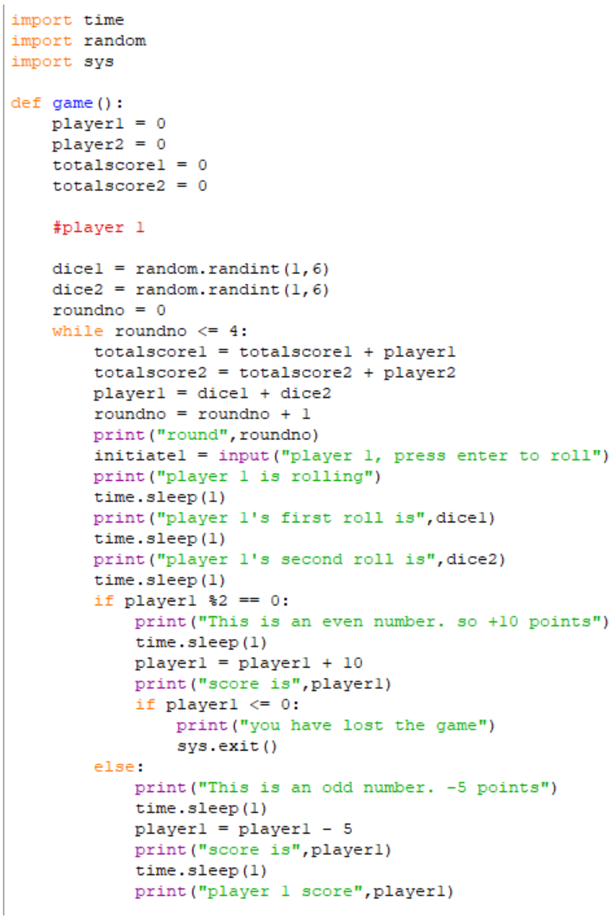
For the high scores function I struggled a lot and failed to make it work many times and could not get it to work in the time given. It prints the username and their own highest score. I then went and printed the username for who has the highest score as it is required in my success criteria.



```
def highscores():
    print(Username1,"your highscore is",totalscore1)
    print(Username2,"your highscore was",totalscore2)
    if totalscore1 > totalscore2:
        print(username1,"you have the highest score")
    else:
        print(username2,"you have the highest score")
```

Game:

For the game function I began by introducing the players and their starting score, I then adds the random number generator to the variables dice1 and dice2. For each round it increments 1 to a variable ‘roundno’. I used a loop so if the ‘roundno’ is less than five then the loop is repeated. I then imported a time function so in-between different code a certain amount of time is taken before the next line is completed. This is all repeated until five rounds have been complete. I struggled trying to get the program to repeat the correct number of round but in the end, I managed to overcome this issue.



```
import time
import random
import sys

def game():
    player1 = 0
    player2 = 0
    totalscore1 = 0
    totalscore2 = 0

    #player 1

    dice1 = random.randint(1,6)
    dice2 = random.randint(1,6)
    roundno = 0
    while roundno <= 4:
        totalscore1 = totalscore1 + player1
        totalscore2 = totalscore2 + player2
        player1 = dice1 + dice2
        roundno = roundno + 1
        print("round",roundno)
        initiate1 = input("player 1, press enter to roll")
        print("player 1 is rolling")
        time.sleep(1)
        print("player 1's first roll is",dice1)
        time.sleep(1)
        print("player 1's second roll is",dice2)
        time.sleep(1)
        if player1 %2 == 0:
            print("This is an even number. so +10 points")
            time.sleep(1)
            player1 = player1 + 10
            print("score is",player1)
            if player1 <= 0:
                print("you have lost the game")
                sys.exit()
        else:
            print("This is an odd number. -5 points")
            time.sleep(1)
            player1 = player1 - 5
            print("score is",player1)
            time.sleep(1)
            print("player 1 score",player1)
```

Testing:

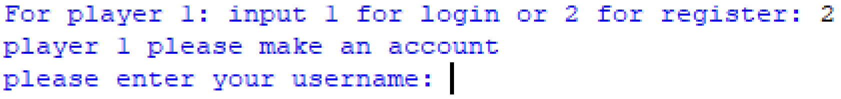
Test table:

Test number	Test type	Test description against user requirements	How the test is performed	Expected result	Actual result (pass/fail)	Further action needed
1	valid	Test whether the menu takes you to the login or register menu after the integer linked to a certain process is inputted.	The user inputs 1 or 2	Login or register menu	pass	no
2	Erroneous	Test whether the menu accepts letters	The user inputs a letter	Error, please re-input either 1 or 2	fail	Yes, I need to add in validation.
3	Valid	Is username in the file	The user enters a username then a print statement is used to see the contents of file	The name should be outputted	pass	no
4	Valid	The incorrect password entered	The user enters a password for the username, so the programme searches through the file for the password.	A print statement is outputted saying "incorrect password, please try again."	pass	no
5	Valid	To test what happens when the correct password is inputted	The user enters a password for the username, so the programme searches through the file for the password.	It logs the user in and asks player 2 to either login or register	pass	no

6	Boundary	To test whether the boundaries on the dice range from 1 to 6	The output of the result from the rolled dice	An integer between 1 and 6 is outputted	pass	no
7	Valid	If the total on the dice odd 5 points are subtracted from score.	The programme outputs an odd total.	Score = 0 because score cannot go below 0	pass	no
8	Valid	If the total on the dice even 10 points are added to score.	The programme outputs an even total.	Score = 10 as score was 0 at the start.	pass	no
9	Valid	Does the score go below 0	The programme runs the game and outputs the score.	Score does not go below 0.	pass	no

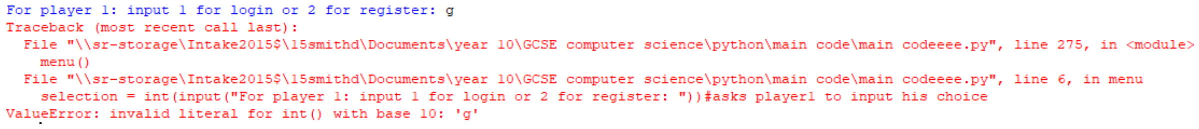
Evidence:

1 –



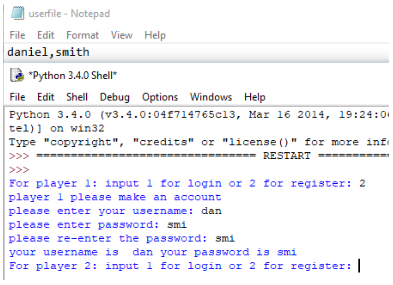
```
For player 1: input 1 for login or 2 for register: 2
player 1 please make an account
please enter your username: 
```

2 –



```
For player 1: input 1 for login or 2 for register: g
Traceback (most recent call last):
  File "\\sr-storage\Intake2015$\15smithd\Documents\year 10\GCSE computer science\python\main code\main codeeee.py", line 275, in <module>
    menu()
  File "\\sr-storage\Intake2015$\15smithd\Documents\year 10\GCSE computer science\python\main code\main codeeee.py", line 6, in menu
    selection = int(input("For player 1: input 1 for login or 2 for register: "))#asks player1 to input his choice
ValueError: invalid literal for int() with base 10: 'g'
```

3 –

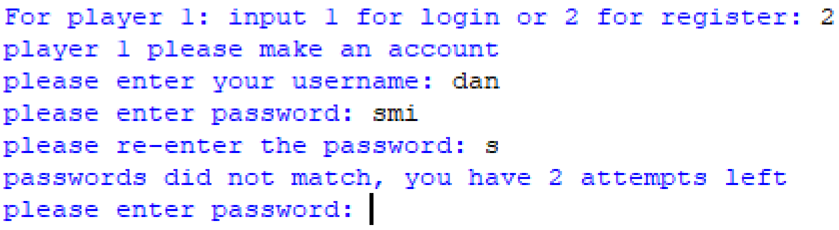


userfile - Notepad
File Edit Format View Help
Python 3.4.0 Shell
File Edit Shell Debug Options Windows Help

```
daniel,smith
```

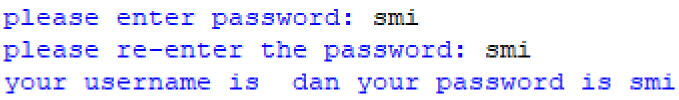
```
Python 3.4.0 (v3.4.0:04f714765c13, Mar 16 2014, 19:24:0
tel)] on win32
Type "copyright", "credits" or "license()" for more inf
>>> ================================ RESTART ==========
>>> 
For player 1: input 1 for login or 2 for register: 2
player 1 please make an account
please enter your username: dan
please enter password: smi
please re-enter the password: smi
your username is  dan your password is smi
For player 2: input 1 for login or 2 for register: 
```

4 –



```
For player 1: input 1 for login or 2 for register: 2
player 1 please make an account
please enter your username: dan
please enter password: smi
please re-enter the password: s
passwords did not match, you have 2 attempts left
please enter password: 
```

5 -



```
please enter password: smi
please re-enter the password: smi
your username is  dan your password is smi
```

Evaluation:

- I found this project challenging but rewarding, because I felt as if I learn a lot about the process of writing a piece of code. I never really had a strong understanding of python however I trialled and failed many times when producing my flowcharts and code. The pseudo code was the easier of the processes as I just followed the flowchart structure.
- I made a mistake in the flowcharts at the beginning as I started to create one long flowchart, which covered every part of the code. I then realised that I needed to use functions as it made it a lot easier to understand. This also helped when writing the pseudocode as I followed the flowcharts patterns as much as possible.
- Writing the code was my hardest challenge as I struggled to convert the pseudocode into python language, which affected me and slowed my progress down heavily. I do not have a fully working dice game due to I struggled to produce working code for the high scores and then ran out of time. If I had more time I may have been able to produce a working high scores table that got the results form the extern al file.

Reference	How it helped
Computing.outwood	
Word	I used this to draw out my flow charts and write my pseudo code.
Stack overflow	Helped me understand how to write code to open, read and write to a file. It also gave me the idea to introduce a time gap between lines of code in the game function.
YouTube	Assisted me when writing the register function
Craig and Dave	Helped me with writing different types of loops