Assignment 1 – Pass the Pigs

Mateo Garcia

CSE 13S – Spring 2023

Purpose

This program simulates a simplified version of the Pass the Pigs game by David Moffat. In the game, players roll a dice to earn points until one of them reaches over 100. The computer outputs a predetermined simulation of the game based on the number of players and the seed. The seed is just a number, and so is the number of players, and both are provided by the user.

How to Use the Program

Running the program will allow the terminal to prompt the user with the line `"Number of players (2 to 10)?"` The user should enter a number between 2 and 10 and hit enter. Following this, the terminal will print `"Random-number seed?"` The user can input a random number, and if it is not a valid seed, a default seed will be generated. The program will then simulate the game, printing the process and the results.

Program Design

The program will have data structures of int, unsigned, array, and enum. The main program is structured with one while loop and increases player's scores based on the dice roll value. The seed is used to generate predetermined random numbers directing to the indexes of the pig array which holds the values to be added to the player's score. The player's turn ends when they roll a zero, and the index of the player names and scores is incremented to move onto the next player. When a player reaches or passes 100 score, it will break out of the loop and print the winner.

Data Structures

Int and unsigned data types would contain the two user inputs, the number of players and the seed. They would also be used for the random function and index values. The arrays would be used for storing the player's scores, as well as the positions of the pig rolled from the dice. And enum would be used for the values of the dice rolls to be able to add points to a player's score.

Algorithms

Will use a while loop where the condition is whether a player has reached a hundred points. Inside the loop, the dice will be rolled, the player's score will be updated, and then if the roll is a zero, then the player will be switched to the next player.

```
loop while ( player_score <= 100 ){
    // gets side of pig die from random function
    roll = pig[random() % 7]
    print( dice_roll )
    score += roll
    if ( score >= 100 ) then
```

```
        break out of loop
    if ( roll == 0 ) then
        switch to next player
}
print( winner )
```

Function Descriptions

There will be one main function only.

- The inputs will be the number of players and the seed
- The outputs will be the printed simulation of the game
- The purpose of the main function is to simulate the game "Pass the Pigs", using the inputted number of players and seed.
- The function will also construct the required data structures in order for the loop to work.

Results

The code I created fulfills all the requirements and meets all the criteria for this assignment. There are no errors, but to make it more efficient I would probably go back and use less if statements because as you can see in Figure 1, I used 4 if statements inside my while loop. I would also try to complete it in less lines of code.

```
42      int i = 0;
43      int prev = 0;
44      int running = 1;
45      while (running) {
46          if (prev == i) {
47              printf("%s\n", player_name[i]);
48              prev++;
49          }
50          int roll = pig[random() % 7];
51          scores[i] += roll;
52
53          printf(" rolls %d, has %d\n", roll, scores[i]);
54
55          if (scores[i] >= 100) {
56              break;
57          }
58
59          if (roll <= 1) {
60              i++;
61          }
62
63          if (i >= num_players) {
64              i = 0;
65              prev = 0;
66          }
67      }
68
69      printf("%s won!\n", player_name[i]);
                                                    42,1          95%
```

Figure 1: Screenshot of the program.

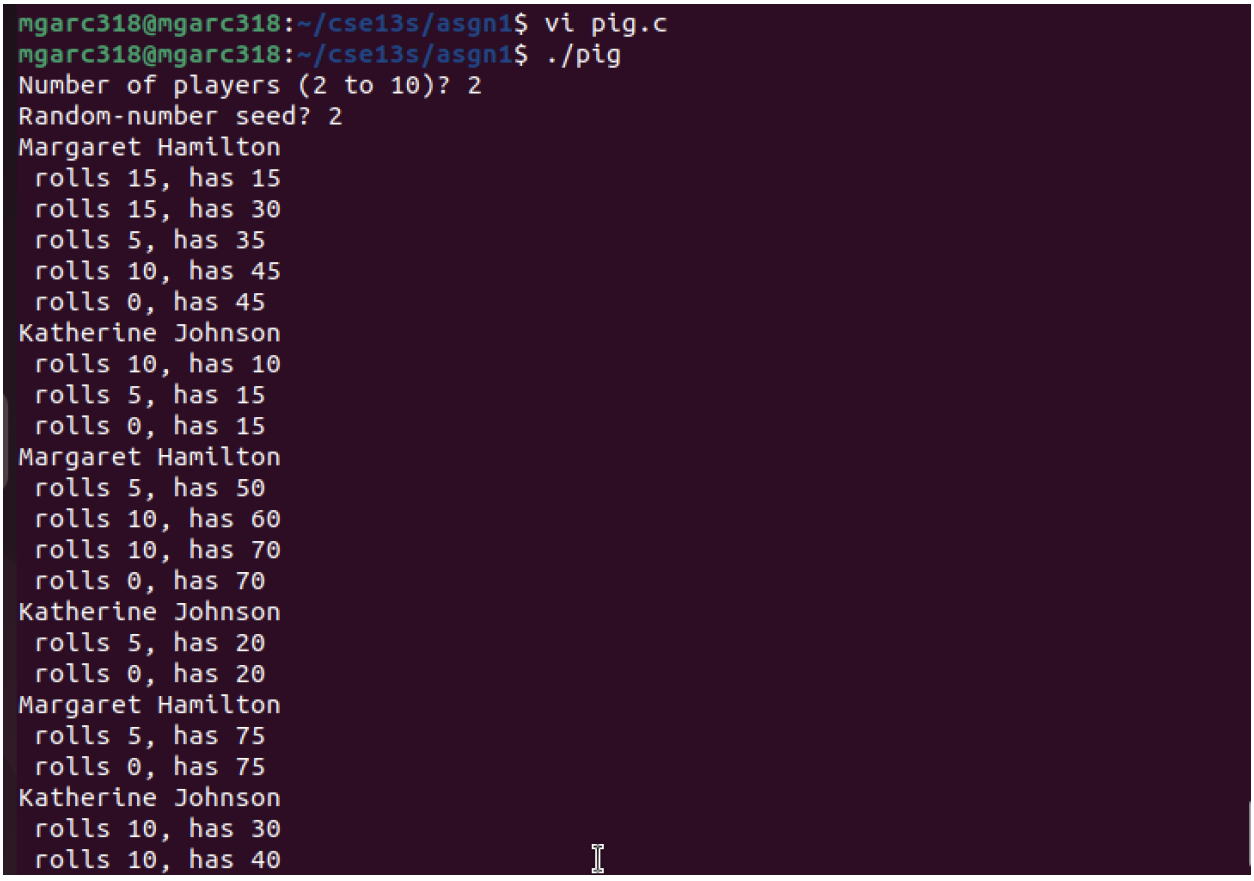

Figure 2: Screenshot of the program running.