Assignment 1 DESIGN.pdf

Description of Program:

This program simulates a turn by turn game of pig with players ranging from 2-10 players and having them roll the pig for points with 100 points being the winning condition. Each player begins their turn with throwing the pig and it can land on SNOUTER, RAZORBACK, JOWLER,TROTTER, SIDE with the number of times the pig can be rolled based on the outcome. If the player rolls a JOWLER the player gain 5 points, RAZORBACK and TROTTER earn 10 points, SNOUTER earns 15 points and rolling a SIDE earns 0 points and the players turn is ended

Files to be included in directory "asgn1":

1. README.md: A file meant to show how to build and run the program and how the program handles bugs
2. Pig.c: The implementation of the program
3. Names.h: The list of the names that will be assigned to
4. Makefile: cleans up the file format tools and makes it more accessible
5. DESIGN.pdf: Describes the entire assignment along with all information regarding the program

Pseudocode / Structure:

```
include name.h file / includes the names in a array required to be accessed by the game
include studio.h
include stdlib.h
enumerate pig positions(SIDE, RAZORBACK,TROTTER,SNOUTER,JOWLER) set each to an int associated with the amount of points

establish constant array type Position with the probabilities of the position associated with the size of array named pig[]

constant function get_position() that translates based on pig position rolled from in to string by using switch function

    return 0

initialize check_winner function
```

```
main
    define int players
    define long long seed
    print "How many players? "
    ask for int input players
    if(players <=1 or players >=11)
        print (" Invalid number of players . Using 2 instead .")
        players equal 2
    print ("Random seed: "
    ask for long long input seed
    if(seed <=4294967295 and seed >= 0)
        set seed through srandom()
        check_winner() function input players
    else
        print(" Invalid random seed . using 2021 instead .")
        set 2021 through srandom
        Check_winner() function input players
    return 0

void check winner function with input as (int players)
    define scores associated with each player as 0 in an array
set win int to 0

WIN:
    while win equal 0
        iterate through players with for loop starting from 0 to players with variable i
            define position as int
            define r as as int
            print((name[i]) rolls the pig...)
            do
                if(scores[i] >= 100)
                    print name[i] wins with scores[i] points!
                    set win to 1
                    goto WIN
                r equals random() mod 7
                position equal pig[r]
                position is added to players score
```

```
          if(r equal 0 or 1 )
               print pig lands on get_position(r)indentation(\n)
          else if r equal 3
               print pig lands get_position(r)
          else
          print pig lands on get_position(r)
     while(position !=0)
```

Notes on Pseudocode and Structure

- typedef enum { SIDE , RAZORBACK , TROTTER , SNOUTER , JOWLER } Position;
  const Position pig[7] = { SIDE , SIDE , RAZORBACK , TROTTER , SNOUTER , JOWLER , JOWLER };
  These lines were provided by Professor Long in Assignment 1 documentation.
-
  get_position(int points): is a function meant to translate the position defined by the pig and translate into string for each case of the int 0-6
- int players: is the number of players used to define many of the parameters of the program and filters out invalid numbers that are noy 2-10
- long long seed: allows for filtering of large invalid seeds that within the bounds of srandom()
- check_winner(int players) is a separate function that takes in the parameter players and prints out the game with each move show eventually ending in the winner
- int scores[10]: sets all players scores to 0 and routinely checks when one of the the scores reaches 100 to print winner
- int win=0 ensures that the while loop will stop the game and end
- int position: records the points associate with the position that the pig lands on
- int r: records random() %7 to have and int with in 0-6 as the cases for the positions