Deeraj Gurram
CSE 13S

Asgn 1 DESIGN.pdf

Description of Program:

This program creates a simple game in which at least 2 and at most 10 players roll a pig shaped die. The pig can land in 5 unique positions, with 2 positions being repeated for a total of 7 positions. Each player starts with 0 points, and continues rolling the pig until it lands on one of its sides. Each position of the pig corresponds to a certain number of points, the Sides are 0 points and end the turn, Razorback and Trotter earn 10 points, Snouter earns 15 points, and both Jowlers earn 5 points. The first player to reach 100 or more points wins.

Files in directory "asgn1":

1. pig.c

   This is the file that contains main() and most of the code that lets the game run.

2. names.h

   This is the header file that contains the ten player names. The names aren't randomized, so names[2] is always John.

3. Makefile

   File that formats code into clang format and compiles pig.c into an executable called pig.

(4) README.md

File that describes how to make and run the program, any known errors, and problems that were encountered while making it.

(5) DESIGN.pdf

Describes how to code the program with pseudocode

Pseudocode:

include names.h — this contains the names of the players

include stdio.h
include stdlib.h
include limits.h
— These contain various functions and values needed throughout my program.

Main:

initialize pig die array {SIDE, SIDE, RAZORBACK, TROTTER, SNOUTER, JOWLER, JOWLER}

initialize playercount to 0
Find number of players by asking for user input
check if user input is valid (input is integer, and is between 2 and 10 inclusive)
Set playercount = 2 if input isn't valid

Initialize unsigned integer SEED
Ask user to input SEED value
Check if SEED is valid (Is positive, unsigned, is a number)

Default SEED to 2021 if input isn't valid
Set random seed to SEED.

Initialize points for each player
Create array of size player count
Loop through array points, and set all values to 0

Initialize main turn incrementor (n)
Initialize points incrementor (k)
Initialize main while loop condition as int condition=1
Initialize nested while loop condition as int minor=1

while (condition == 1)

Initialize nested while loop condition as int minor=1

Create for loop to check points of all players at beginning of each round
for (int h=0; h < player count; h++)
if (points[h] >= 100)
print name of player who won
condition = 0

Initialize variable to contain this round's die position
int position = pig[random() %7]

Print the name of whoever's turn it currently is

Create while loop for each individual turn
while (minor = 1)

End this iteration if pig lands on SIDE

```
if (position == pig[0] or position == pig[1])
    print roll outcome
    minor = 0

else if (position == pig[2] or position == pig[3])
    add appropriate amount of points to points[k]
    print roll outcome
    print point value of this roll

else if (position == pig[4])
    points[k] = points[k] + 15
    print roll outcome
    print point value of this roll

else if (position == pig[5] or position == pig[6])
    points[k] = points[k] + 5
    print roll outcome
    print point value of this roll
```

If this iteration of the while loop ends
increment n by 1
increment k by 1
Restart the loop until a player's points are >= 100.