

1.1 Program Description:

This program runs a version of the dice game *Pass the Pigs*, by David Moffat. This simplified form of the game is played with n number of players, ranging from two to ten players. Arranged in what can be visualized as a circle, the players roll the dice, or the *pig*, in an effort to earn points. Whenever the pig lands on its side, the next player's turn commences. The first player to reach 100 points wins the game.

1.2 Files Included in the Directory:

- **pig.c**
 - The source file contains the main code that allows the game to run.
- **names.h**
 - Header file that contains an array of 10 elements, containing various names that are used as the names of the players of the game.
- **Makefile**
 - The file contains the tasks required to compile the program from the source code.
- **README.md**
 - The text file is written in markdown syntax that describes the process to build and run the program. In addition, it also describes error handling and any bugs and problems that were come across in development.

1.3.1 Pseudocode and Structure:

Get number of players through user input

if the number of players, n , is not $2 < n < 10$, then assume 2 players

Get random number generator seed

if the seed number is invalid (exceeds `UINT_MAX`) assume seed number 2021

initialize random number generator

while points < 100:

start at player 0:

generate random number

if random number corresponds with side

increase player array indice by 1

if random number corresponds with razorback OR trotter

add 10 points to player points

if random number corresponds with snouter

add 15 points to player points

if random number corresponds with jowler

add 5 points to player points

if points \geq 100

print player wins prompt

1.3.2 Program Explanations:

1. User inputs:
 - a. The user inputs are tied by conditionals to only accept inputs in the defined valid range (2-10 players, and the 32-bit limit of an unsigned integer for the random number generator (RNG) seed ($2^{32} - 1$))
2. The actual game:
 - a. With each “roll,” a new number is generated by the RNG.
 - b. Each generated number corresponds to an element of the enumerated array that indicates what position the pig landed in (the player’s roll).
 - c. The game continues until a player reaches 100 or more points, at which point a message announcing the player’s victory is outputted.

1.4 Additional Credits:

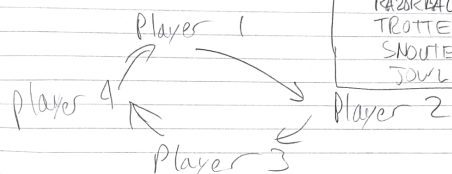
- I attended Sloan’s section at 1:15 PM on Wednesday to learn more about creating a Makefile for reference in future assignments.
- In addition, Sloan also provided the limit for the SEED in section, which is `UINT_MAX`.
- A pig position array from `asgn1.pdf` was used for enumerating the pig positions.

1.5 Error Handling:

1. The failure to update elements of the points keeping array.
 - a. Solution: After attempting a few methods to change the outcome (usage of arithmetic operators instead of assignment operators, initializing the array in different ways), the bug was solved by initializing the array in the form of `int array[elements] = { 0, ..., 0 }` in combination with using the following method to update elements of the array: `array[element] = array[element] + points`.
2. The exact formatting of the whitespace at the end of each output line.
 - a. Solution: Usage of the following diff command format provided on the class discord to compare outputs and adjust print statements accordingly.
 - i. `(diff <(echo 5 2000 | ./pig) <(echo 5 2000 | ../../resources/asgn1/pig) | cat -A)`
3. The player counter overcounting and causing a bit overflow.
 - a. Solution: Addition of an additional conditional to keep the counter within the number of players set by the input.
4. The program is infinitely repeating, and the players rolling the side position every time.
 - a. Solution: Moved the `srand()` command to the outside of the while loop in order to prevent the RNG’s first generated number from repeating infinitely.

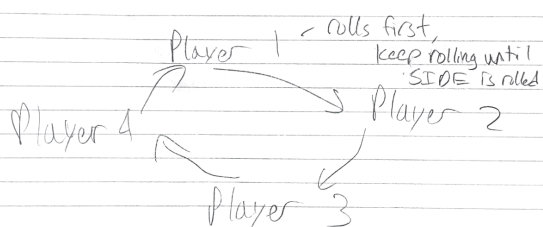
1.6 Supplementary Diagrams:

Pass the pigs



Pig positions

SIDE - end turn
PIGBACK - 10 points
TROTTER - 10 points
SNOUTER - 15 points
JOWLER - 5 points



- First player to reach 100 points wins