Assignment 3 Design Doc

Tanya Gyanmote; cruzid:tgyanmot

10/16/22

Details:

Creating a centuries-old game of chance with a four-sided top known as a dreidel

Implements:

- A spin_dreidel() function returns one of the four letters N, G, H, or S.
- A play_game (int n_players, int coins_per_player, int * n_rounds) functions to play a single game of dreidel and gets back a number corresponding to the player who won.

Pseudocode:

Spin_dreidel(void)

1. Create char variables for G, H, N,S
2. Create uint64 random num and mod it by 4, because you always wanna end up with 0,1,2,3
3. Create if statements to check if the corresponding number is equal to the random number
4. Return the letter according to the corresponding number
    a. G =0, H =1, etc

Play_game (int n_players, int coins_per_player, int * n_rounds)

1. Create an array of players with the number of players
2. Have a variable to keep count of eliminated players
3. Create a variable for the pot, to keep track of the coins in the pot
4. Create a for loop to fill up the array with the number of coins per player

5. Create a while loop to run as the game goes
    a. Create a variable turn, when u spin the dreidel
    b. Create a for loop that runs rounds
        i. Before you spin the dreidel make sure a player doesn't have -1 coins, to make sure they aren't eliminated
        ii. If the turn is G gives the player the pot, then empty the pot
        iii. If the turn is H, then give the player half the pot and subtract half the pot from the pot
        iv. If the turn is S then the player gives one coin to the pot
            1. If a player has 0 coins and lands on S they get eliminated
                a. Make their coin value -1
                b. Subtract one from eliminated players
            2. If my bool for verbose v is true then have a print statement that prints everything a player gets eliminated
        v. Check for the remaining player and if their coin value is -1 go through the players to see who doesn't have -1 coins and return that index
    c. Keep count of rounds += 1

Main (play-dreidal)

1. Define the different options p, s,c,v
2. Initialize list_players and my test_v boolean
3. Create variables for default values and give them the default values
4. Have a while loop
    a. Create cases for p,s,v,c using switch

i. In each case change the input type according to what it should be
    1. Using strtoul for uint64_t type, and atoi for int
ii. Make sure the input values are valid values
    1. For case p make sure the players are 2-8,
    2. For case c make sure that the coins are from 1-20
    3. For case v make sure that the seed is less than 9999999999
        a. For all cases break, otherwise if they don't meet the criteria then return 1

5. Create a variable for *n_rounds set to 0
6. Then set the seed
7. Create a variable for the winner player index while calling play game using converted values from the above
8. Print according to the format, with winner, player number, coins per player, rounds, and seed value

Files:

1. Dreidel.c: This file contains the function spin_dreidel() and play_game()
2. Play-dreidel.c: This file will call my functions from dreidel.c
3. Dreidel.h : This file contains definitions for functions defined in dreidel.c and used by other code
4. Makefile: This file will allow the grader to type make to compile your program.
5. README.md: This file will describe how to build and run my program and list the command-line options it accepts and what they do.

6. DESIGN.pdf: Describe the purpose of your program and communicate the overall design of the program with enough detail

7. WRITEUP.pdf: Discussion of the results of my tests.

8. Mtrand.h: Provided in asgn3-files.tgz.

9. Mtrand.c: Provided in asgn3-files.tgz.