Michael Nguyen
mnguy181@ucsc.edu
1/17/2021

CSE13S Winter 2021
Assignment 1: The Garlic Game
Design Document

PURPOSE:

The Garlic game is a game in which 2-10 vampires stand in a circle and start with 3 lives each. The game then begins where in each round, each vampire rolls 2 die.
During a round:

- A vampire will roll 2 die if they are alive, going in a sequence of the circle. Only vampires with 1 or more lives can roll. Even if a vampire has 0 lives, they have a chance to be resurrected later in the game.
- The vampire that rolls the lowest number loses a life by eating garlic. If two or more vampires roll the same lowest number, the first vampire that rolled it must eat the garlic.
- Rolling two 6's results in a midnight. When a midnight is rolled, the vampires on the left and right of the vampire that rolled the midnight resurrect or sparkle. If the vampire (on the left or right) has 0 lives, then they are resurrected. If the vampire (on the left or right has more than 0 lives (or in other words, alive), then they sparkle. In both cases, the vampires on the left and right gain 1 life. Nothing happens for the vampire that rolled the midnight.
- The garlic game ends if there is only one vampire left with any amount of lives and is declared as the winner of the Garlic Game.

The purpose of the program is to serve as a simulation for the Garlic Game in which the program accepts user input for the number of vampires/players and the random seed and proceeds to print the output of the different rounds and the situations that arises (including what each vampire rolled, which vampire eats the garlic, the midnight occurrence [who gets resurrected/sparkled], and the winner of the game). The program will end when a winner is declared.

TOP LEVEL LAYOUT/STRUCTURE:

The following pseudocode covers the overall layout and structure of my program.

A) Include header file and libraries
B) Function declarations
    - Function roll() will return [0,5]
    - Function left() returns position left of current player
    - Function right() returns position right of current player
C) Main:
    - Read program arguments for number of players and random seed
    - Error check number of players and random seed
    - OUTER LOOP for rounds exiting when one player left:
      - Print out the current round
      - INNER LOOP for each players rolls
        - Roll 2 die

        - Check midnight condition
        - Print out what the player rolled
    - INNER LOOP to find lowest roll
        - Print out the player that eats garlic
    - Deduct life from the vampire that eats garlic
- Print out the winner of the game

A) Include header files and libraries

In this section of my code, I will need to include the header file names.h which has been supplied to me by the Professor and the TAs from the git resources section. In addition, I will also need to include libraries stdio.h , stdlib.h , inttypes.h . These libraries will supply functions such as random() and srandom() which are needed for the rolling of the two dice during the game. In addition, srandom() is needed to generate the random seed's starting point. The inttypes library is used for error checking of the random seed user input and is also used to set the boundaries for the random seed to ensure it is no greater than 32 bits and is not a negative number.

B) Function declarations

In this section of my code, I will need to make a function known as roll() that essentially returns random() modulo 6. This will be used to generate a number from 0 to 5 inclusive and serves to simulate the rolling of a single dice. Although we may be used to rolling a 1-6 dice, 0 to 5 is used to access a 2-D array provided in the header file names.h known as rolls. This array will be used when printing out the specified names of the rolls during each round.

Two other functions left() and right() essentially return the position of the left and right vampires whenever a midnight is rolled. The function requires 2 parameters which include the current position of the player that rolled the midnight and the number of players in the game.This function is used to give 1 life to the left and right vampires of the player that rolled a midnight. These two functions have been provided by the program document.

C) Main

- Reading user input and error checking

  In this section of my code, I will use scanf() to scan the integer value provided by user input after prompting the user for the number of players they wish to play the game. The game can only be played with 2-10 players. I will then Error check for proper input. If the input is invalid, then I will use exit(1) to exit the program safely.

  Next, I will prompt the user for random seed. Using scanf, I will read the user's input, error checking to ensure that the integer is no greater than 32 bits and is not a negative number.
- OUTER LOOP for rounds exiting for when one player is left alive

  In this section of my code, I will first initialize a counter for the number of players alive, setting it initially to the number of players scanned from the program arguments. Then I will use a while() loop that serves as the continuation of the game, exiting when the current number of players left alive is 1.

  First, I will use the printf() function to print out the round number.

Then, I will initialize a for() loop that starts from 0 and ends at the number of players. This loop's purpose will be to iterate through each player's rolls during the round. In this for() loop, I will begin by using the roll function defined above to roll 2 dice, storing their values into a variable while also storing the sum of the rolls into an array holding the rolls of the players. Then, I will check if these rolls are both 5s (essentially checking for the midnight condition). If a midnight has indeed occurred, I will add 1 life to the right and left players, either resurrecting them or having them sparkle. Otherwise, if a midnight did not occur, I will printf() the current player's name and use the rolls array in the header file to print out what they rolled.

At the end of each round (after the for loop is done), I will iterate through an array that stored the round's rolls and find the minimum. This iteration will be done with another inner for() loop. When the minimum value is found, I will store the index of the minimum value in the array, and use this index to print out which player was forced to eat the garlic and deduct 1 from their lives.

If the player that was forced to eat the garlic has more than 0 lives, I will print out how many lives they have left. Otherwise, I will print out that they died at the end of the round. This will be done using the conditional if() statement in my program.

- Printing out the winner of the game
  After the while() loop has exited, I will print out the winner of the game which is whichever player has more than 0 lives.

UPDATES:

I needed to create an array that stored the lives of each player. This will be iterated through at the end of each round to deduct lives from a corresponding index of a player in the array. In addition, It will need to be used to find the person with the most lives at the end to find the winner. The iteration for the winner was done using a for() loop.

I also needed the lives array for the midnight occasion in order to add 1 life to the left and right vampires of the vampire that rolled the midnight. This iteration was done using a for() loop.