Ramos, Ron Isaiah

Velasco, Gimel

Computer Science 125 Documentation

1. **Lottery Scheduling**

Lottery scheduling is a randomized resource allocator. Rights for this resources are represented as tickets. The winning ticket gets the resource.

1. **Data Structure**

The data structure we used in this implementation is a doubly linked list. Under each node of the linked list we have the process ID number, tickets, burst time, process time, next client node, and previous client node.

1. **Functions**

Here are the functions we used to create the program:

* 1. list\_insert – this function inserts a process in the doubly linked list.
  2. print\_clientlist – this function prints the process ID, tickets and burst time of all the processes.
  3. get\_totalTickets – this function gets the total number of tickets.
  4. getrun\_winnerTicket – this function is implements the lottery scheduling mainly.
  5. srch\_process – this function searches for a process using its ID number.
  6. del\_process – this function deletes a process.
  7. list\_deleteAll – this function deletes all processes.
  8. srch\_ticket – this is used for checking whether a ticket is already used by another process.

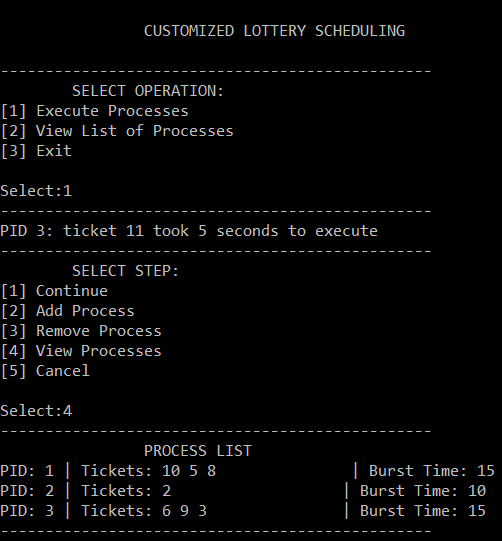
1. **Difficulties Encountered**

There are a lot of challenges that we encountered in this project. Firstly, the representation of the data structure using C in this implementation was difficult. Linking the nodes without the loss of data was also a challenge. Of course, the efficiency of the program for the user is always in mind, and should always be a challenge. The GUI was difficult in a way that it should be ergonomic and user-friendly.

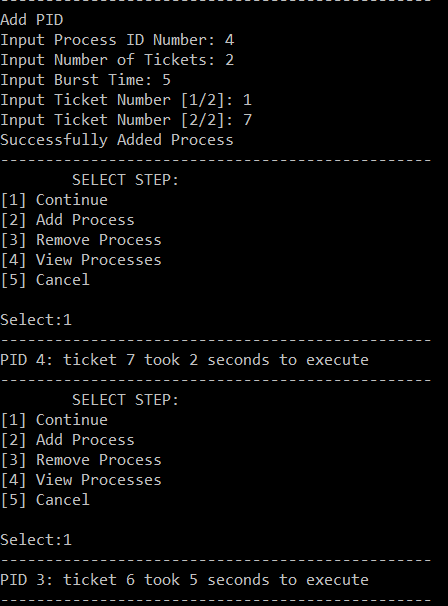
1. **Recommended Improvements**

We recommend a user-defined maximum tickets and maximum tickets of a process. Also, using this algorithm, managing memory can expand the usage of this implementation.

1. **Screenshots**
2. Starting the program with this interface, and viewing available processes

. 

1. Adding processes and executing



1. Deleting processes

