Birla Institute of Technology and Science, Pilani Hyd Campus
BITS F232: Foundations of Data Structures and Algorithms
1st Semester 2023-24: Assignment 1 (Set C)
Maximum marks: 20   Date of Submission: 8th Nov 2023

Q.1 Problem Statement: Memory Management Program with Two Queues

Problem Description:

Implementing a memory management program that uses two queues, one for allocated memory and another for free memory. The program should efficiently allocate and deallocate memory blocks of varying sizes.

Input:

The program takes the following inputs:

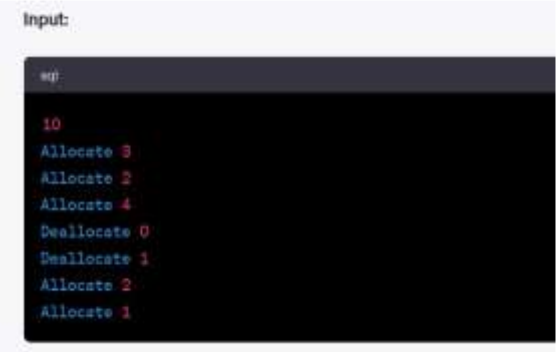
- An integer memorysize (1 <= memorysize <= 1000) representing the total size of available memory.
- A series of memory allocation and deallocation requests, each described by a line containing one of the following operations:
  - Allocate <size>: Allocate a memory block of the specified size, where size is an integer (1 <= size <= memorsize).
  - Deallocate <blockIndex>: Deallocate the memory block with the specified index, where blockIndex is an integer representing a previously allocated memory block.

Output:

For each memory allocation operation, print a message indicating the allocated memory block's range (start and end indices) and size. For each memory deallocation operation, print a message indicating the deallocated memory block's range.

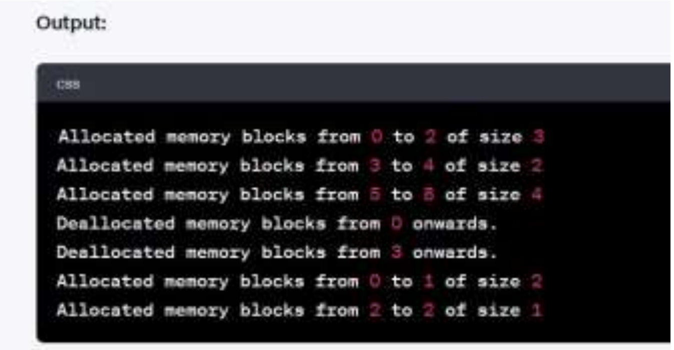
If a memory allocation request cannot be satisfied due to insufficient available memory or an invalid size, print a corresponding error message.

Example run:



Input:

```
10
Allocate 3
Allocate 2
Allocate 4
Deallocate 0
Deallocate 1
Allocate 2
Allocate 1
```



Output:

```
Allocated memory blocks from 0 to 2 of size 3
Allocated memory blocks from 3 to 4 of size 2
Allocated memory blocks from 5 to 8 of size 4
Deallocated memory blocks from 0 onwards.
Deallocated memory blocks from 3 onwards.
Allocated memory blocks from 0 to 1 of size 2
Allocated memory blocks from 2 to 2 of size 1
```

(Note: Part program is given in file 3.1)

Q.2 In a top-secret mission, Team 1 is tasked with infiltrating enemy lines to extract vital intelligence. Each operator submits their call sign, a classified ID, and sensitive data. The program stores these details in a secure vault and analyzes prime factors, while Team 2 prepares to extract critical information. Upon their return, Team 1 faces a stringent authentication process. Success means mission accomplished; failure is the gravest sacrifice. The program prints the verdict, and if all authentication checks succeed, it issues a resounding salute to the team for securing the lifeblood of the operation, ensuring victory on the battlefield.

Operation Team (Team 1):

The military major should prompt the Team 1 leader to enter the number of Operation Team members (numMembers).For each operation program should prompt the Team member to enter their name and a unique ID (integer).After gathering this information, the program should allow each team member to give input as an integer value representing secret information. The program should store the secret information for each team member in a stack (gatheredData).For each secret information value entered, the program should find its factors and store prime factors in a queue (collectedData).

Central Team (Team 2):

Team 2 should retrieve data from Team 1's queue (collectedData).It should then find and store the second-highest prime number from the retrieved prime factors as a vital information.

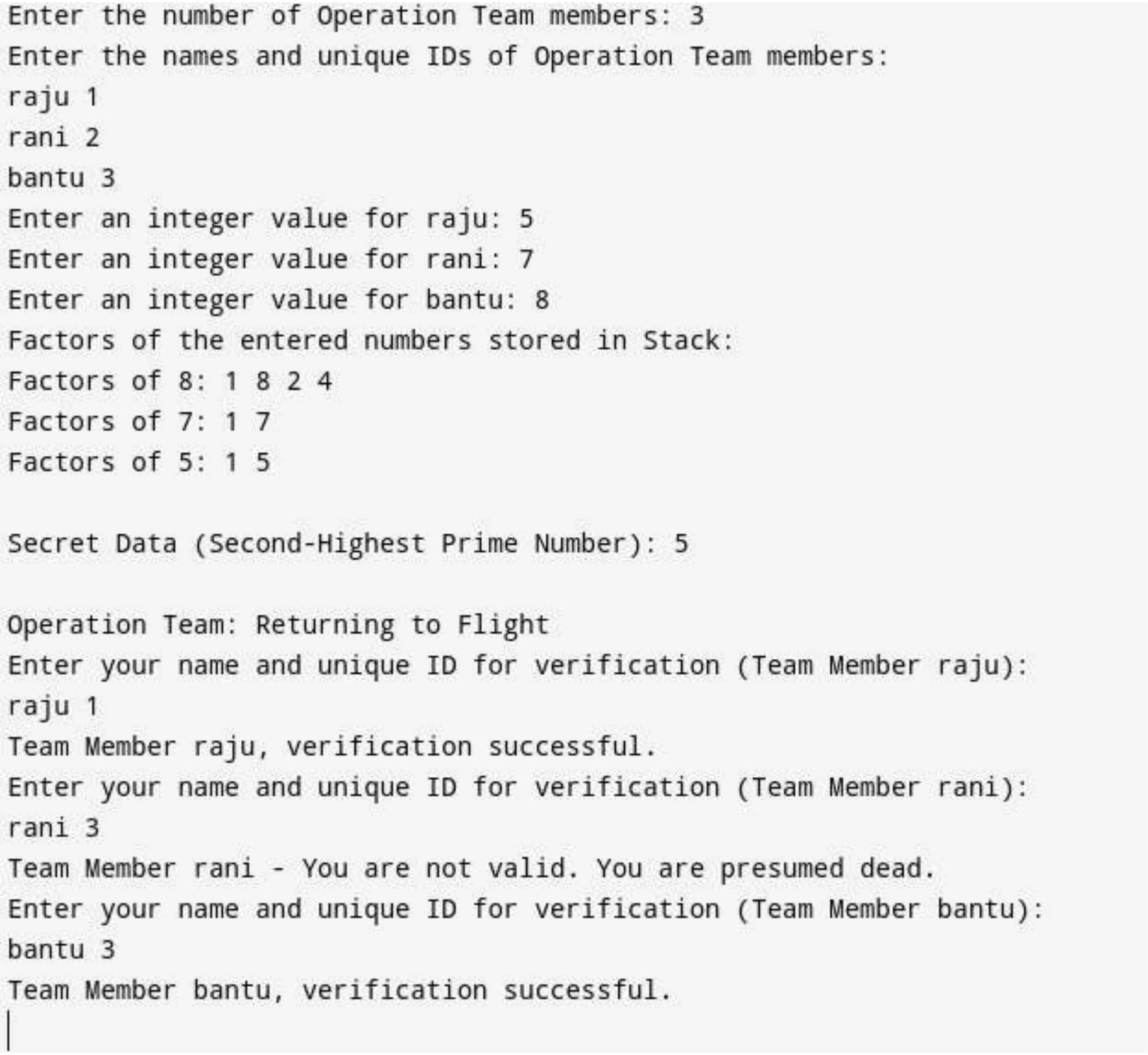
Verification and Conclusion:

After successfully retrieving data, Team 1 returns to the flight.Each Team 1 member must provide their name and unique ID for verification.If the provided name and ID match the stored information for that member, they are considered valid and have achieved their mission.If any Team 1 member provides incorrect name or ID, they are presumed dead.

Finally, the program should print whether all Team 1 members are valid or not. If all are valid, it should display a success message.

(Note: Part program is given in file 3.2)

Sample Run:



```
Enter the number of Operation Team members: 3
Enter the names and unique IDs of Operation Team members:
raju 1
rani 2
bantu 3
Enter an integer value for raju: 5
Enter an integer value for rani: 7
Enter an integer value for bantu: 8
Factors of the entered numbers stored in Stack:
Factors of 8: 1 8 2 4
Factors of 7: 1 7
Factors of 5: 1 5

Secret Data (Second-Highest Prime Number): 5

Operation Team: Returning to Flight
Enter your name and unique ID for verification (Team Member raju):
raju 1
Team Member raju, verification successful.
Enter your name and unique ID for verification (Team Member rani):
rani 3
Team Member rani - You are not valid. You are presumed dead.
Enter your name and unique ID for verification (Team Member bantu):
bantu 3
Team Member bantu, verification successful.
|
```
