

# CS69011: Computing Lab-1

## Test-1

July 24, 2024

### ===== Instructions =====

1. Q1 and Q3 carry 35 marks each, and Q2 carries 30 marks.
2. In the case of user input assume only valid values will be passed as input.
3. You can use C or C++ as the programming language. **However, you are not allowed to use any STL libraries in C++**
4. Regarding Submission: For each question create a separate C file. -> <rollno>\_Q1.c, <rollno>\_Q2.c, <rollno>\_Q3.c. Create a zip file of all these C files in the name <rollno>\_T1.zip and submit it to Moodle. For example, if your roll number is 24CS60R15, then your file names will be 24CS60R15\_Q1.c, 24CS60R15\_Q2.c, 24CS60R15\_Q3.c and your zip file name will be 24CS60R15\_T1.zip.

### =====

1. Pedro, the leader of the Mafia “Maltese Falcon”, was saddened by the untimely murder of his Caporegime Vito by the rival gang “Morello Salieri”. In order to exact revenge, Pedro captured N members of the rival gang. He killed (N-1) members by gunshot sparing 1 to extract information about “Morello Salieri”. He did so in the following manner:

All the N members are arranged in a circular fashion 1, 2, ..., N. After the arrangement and starting from the 1st person, Pedro kills every  $K^{\text{th}}$  member (starting and including from the first position) ( $K \leq N$ ) in the circle until only one person remains. The goal is to find out which position the last remaining member occupies. Print the answer in the terminal. [35]

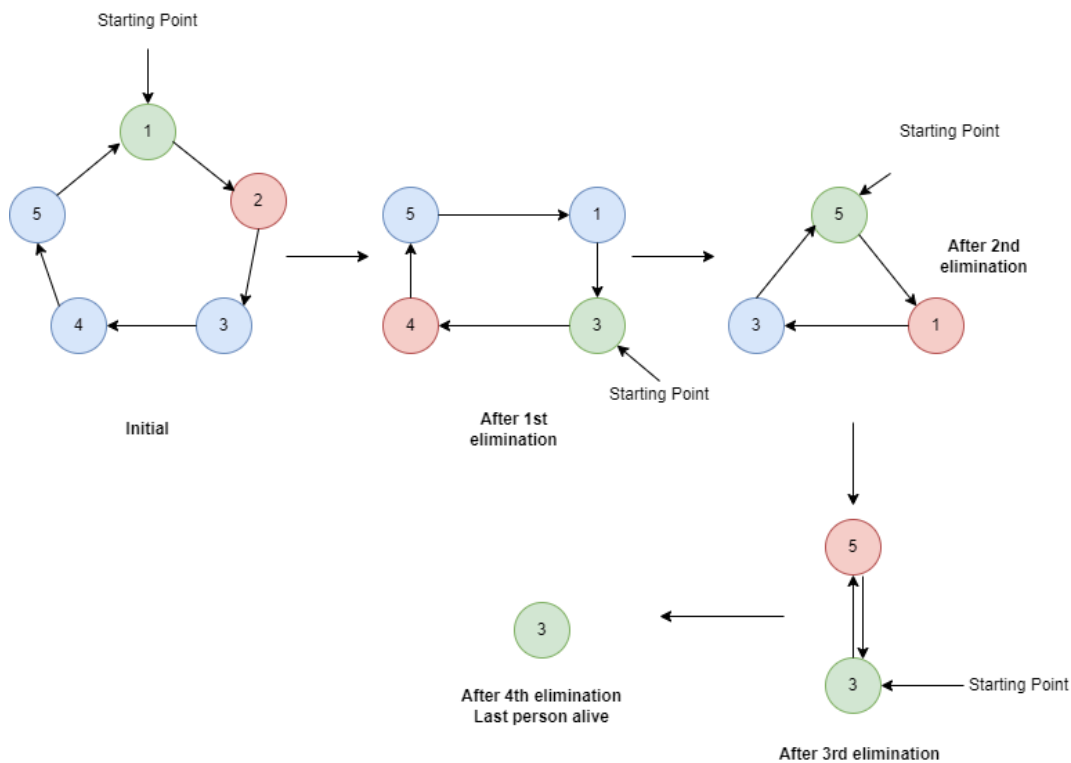
[Note:  $K=2$  fetches 30% marks &  $K=$  any Natural Number  $\leq N$  fetches full marks]

Test case #1:  $N = 5$  and  $k = 2$  (see figure)

Output: 3

Explanation: First, the person at position 2 is killed, then the person at position 4 is killed, then the person at position 1 is killed.

Finally, the person at position 5 is killed. So the person at position 3 survives.



Example for N = 5 and K = 2

Test case #2: N = 7 and k = 3

Output: 4

Explanations: The persons at positions 3, 6, 2, 7, 5, and 1 are killed in order, and the person at position 4 survives.

Hint: Use Arrays/Linked List if required.

- In a faraway land, there exists a mystical island where a great treasure is hidden. The treasure can only be unlocked by solving a unique mathematical puzzle that the island's guardian presents to every treasure seeker.

The guardian, a wise old sage, gives you a list of integers (at least five integers). Your task is to find any five distinct integers from the list so that the product of those numbers is maximum. Print the answer(the maximum product) in the terminal.

In simple words, for an array (of size  $n \geq 5$ ) comprising integer elements (positive, zero or negative and may contain duplicates), you need to find such 5 elements whose product is maximum among the elements chosen.

Are you ready to take on the challenge and unlock the ancient treasure?

[Hint: If You Need To Use Sorting, Use **Merge Sort**. Using Sorting Algorithms Other Than Merge Sort Will Lead To **25% Deduction** of marks.]

Testcase #1

Input : n=5    array=[-1, -2, -3, -4, -5]

OUTPUT: -120

Testcase #2

Input: n=11    array= [1, 0, 5, 4, -4, 3, 1, 1, 0, -3, 2]

OUTPUT: 720

Testcase #3

Input: n=6    array=[-1, 0, 0, 0, -1, -1]

OUTPUT : 0

Testcase #4

Input n=10    array=[-23,12,34,-1,2,3,1,1,2,10]

OUTPUT: 93840

Testcase #5

Input n=10    array=[-23,12,34,1,2,3,1,1,2,10]

OUTPUT: 24480

- 
3. Write a C program to create a Queue Data Structure using a **Circular Linked List** using **one pointer as front and rear**. Do the following :
- A struct Queue which contains a value of type int and a pointer to the next Node.
  - A function Enqueue(Queue \*q, int x) which inserts value x at the rear of the queue
  - A function Dequeue(Queue \*q) which deletes a value from the front of the queue. Show the empty error with an appropriate msg.
  - A function Peek(Queue \*q) which checks the front element of the queue.
  - A function Size(Queue \*q) which returns the size of the queue (returns int value).
  - A function Print(Queue \*q) which prints the Queue contents.

Create a main() function in a menu-driven format such that operations b-f mentioned above can be performed.

Using **other methods [like using array] of implementing a queue apart from the one mentioned** will incur a **penalty of 20% of the total marks**. [6\*5+5 =35]

Example :

=====Terminal Interaction Begins here=====

Queue Menu :

1. Enqueue
2. Dequeue

3. Peek
4. Size
5. Print
6. Exit

Enter your choice : 1

Enter the value : 10

Value Inserted : 10

Queue Menu :

1. Enqueue
2. Dequeue
3. Peek
4. Size
5. Print
6. Exit

Enter your choice : 3

Peeked element : 10

Queue Menu :

1. Enqueue
2. Dequeue
3. Peek
4. Size
5. Print
6. Exit

Enter your choice : 2

Deleted Element : 10

[Note : When choice is 6, code stops execution else it does the operations described.]

---