VARUN BHARGAVA – 241010282
DATA STRUCTURES TASK-8

Task 01: Josephus Problem:

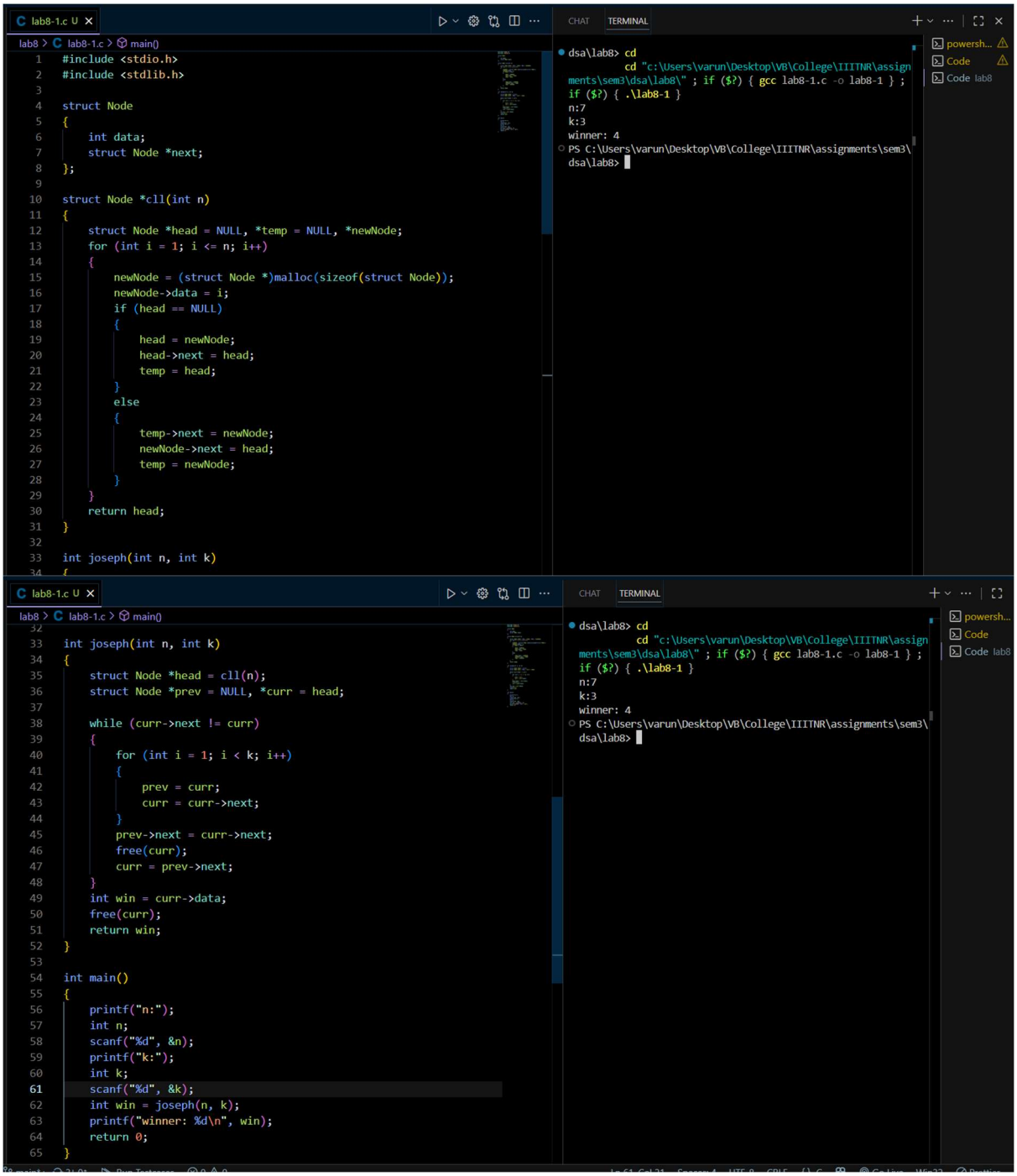
(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab8/lab8-1.c)

Write a program that simulates the Josephus Problem, where n players stand in a circle and every kth player are eliminated until only one player remains. The program should determine and print the position of the winner (survivor).

Example:

Input: n = 7, k = 3

Output: The winner is player 4



Task 02: Single Linked List – Longest Ascending Sequence:

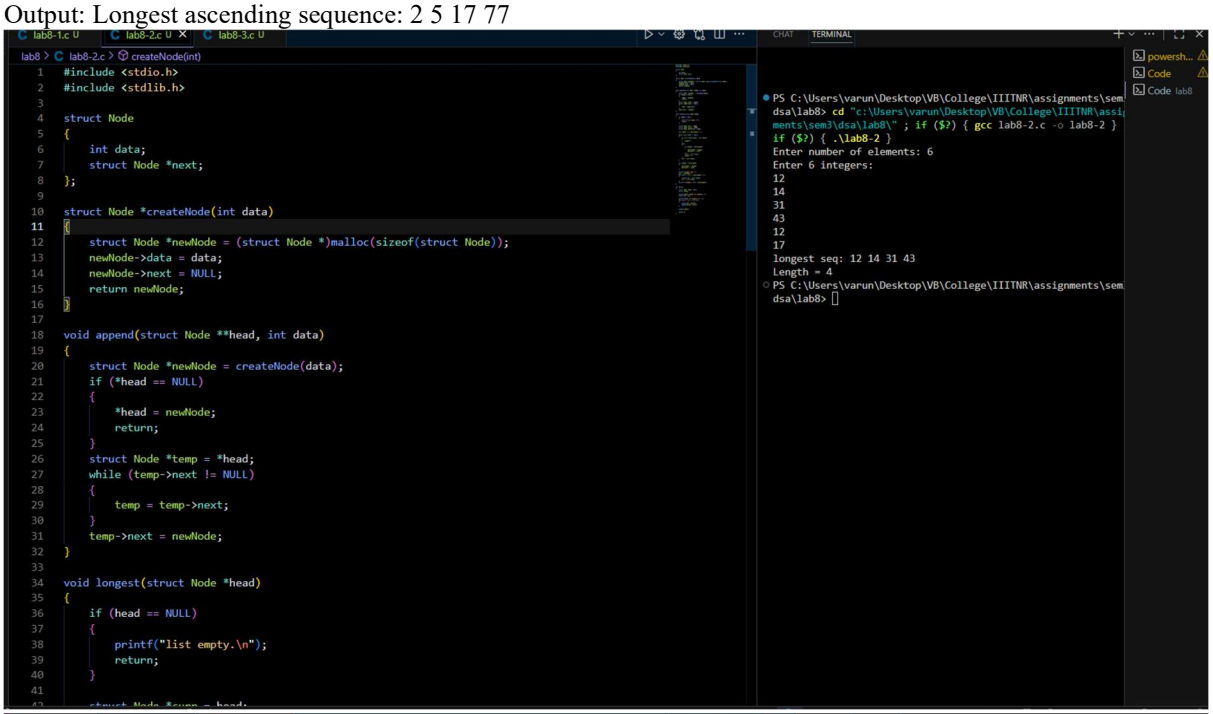
(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab8/lab8-2.c)

Write a program using a single linked list that takes input from the user and prints the longest sequence of ascending order integers in it along with the length.

Example:

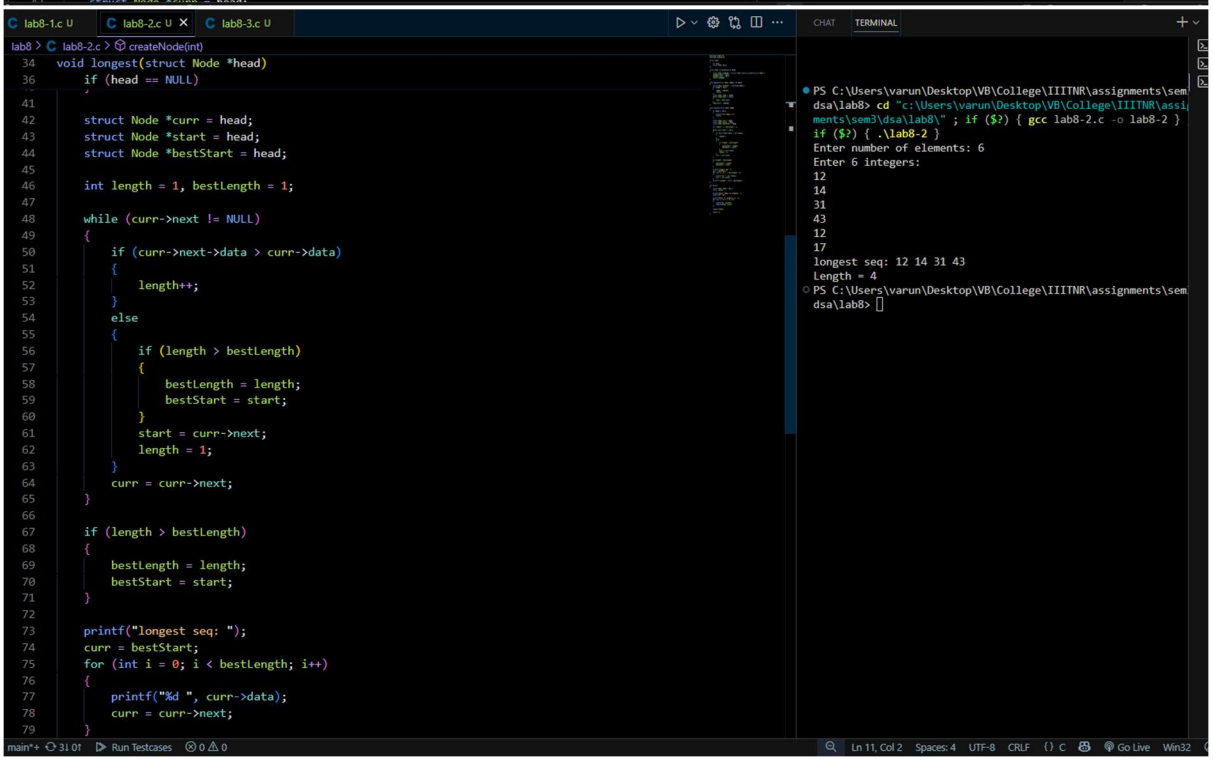
Input: 20, 2, 5, 17, 77, 7, 5, 3

Output: Longest ascending sequence: 2 5 17 77



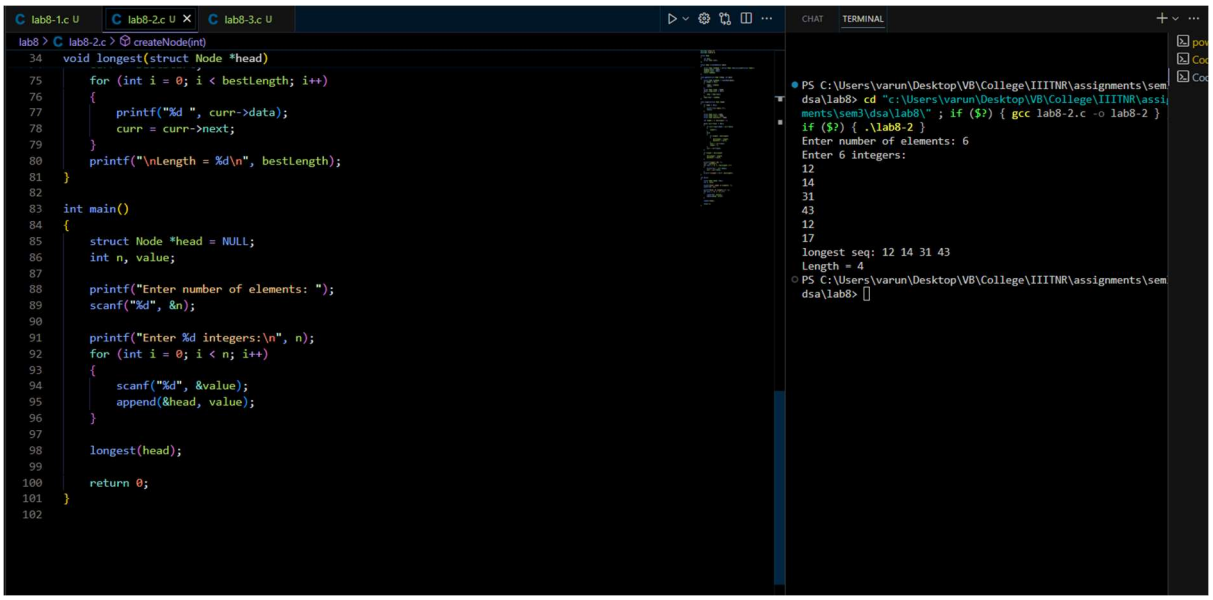
```c
#include <stdio.h>
#include <stdlib.h>

struct Node
{
    int data;
    struct Node *next;
};

struct Node *createNode(int data)
{
    struct Node *newNode = (struct Node *)malloc(sizeof(struct Node));
    newNode->data = data;
    newNode->next = NULL;
    return newNode;
}

void append(struct Node **head, int data)
{
    struct Node *newNode = createNode(data);
    if (*head == NULL)
    {
        *head = newNode;
        return;
    }
    struct Node *temp = *head;
    while (temp->next != NULL)
    {
        temp = temp->next;
    }
    temp->next = newNode;
}

void longest(struct Node *head)
{
    if (head == NULL)
    {
        printf("list empty.\n");
        return;
    }
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem
dsa\lab8> cd "c:\Users\varun\Desktop\VB\College\IIITNR\assi
ments\sem3\dsa\lab8\" ; if ($?) { gcc lab8-2.c -o lab8-2 }
if ($?) { .\lab8-2 }
Enter number of elements: 6
Enter 6 integers:
12
14
31
43
12
17
longest seq: 12 14 31 43
Length = 4
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem
dsa\lab8>
```



```c
void longest(struct Node *head)
    if (head == NULL)

    struct Node *curr = head;
    struct Node *start = head;
    struct Node *bestStart = head;

    int length = 1, bestLength = 1;

    while (curr->next != NULL)
    {
        if (curr->next->data > curr->data)
        {
            length++;
        }
        else
        {
            if (length > bestLength)
            {
                bestLength = length;
                bestStart = start;
            }
            start = curr->next;
            length = 1;
        }
        curr = curr->next;
    }

    if (length > bestLength)
    {
        bestLength = length;
        bestStart = start;
    }

    printf("longest seq: ");
    curr = bestStart;
    for (int i = 0; i < bestLength; i++)
    {
        printf("%d ", curr->data);
        curr = curr->next;
    }
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem
dsa\lab8> cd "c:\Users\varun\Desktop\VB\College\IIITNR\assi
ments\sem3\dsa\lab8\" ; if ($?) { gcc lab8-2.c -o lab8-2 }
if ($?) { .\lab8-2 }
Enter number of elements: 6
Enter 6 integers:
12
14
31
43
12
17
longest seq: 12 14 31 43
Length = 4
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem
dsa\lab8>
```



Task 03: Anagram Check:

(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab8/lab8-3.c)

Check whether two input strings are anagrams or not

Example:

Input: listen, silent

Output: Strings are Anagrams