Rajalakshmi Engineering College

Name: Udhayan Shejay VS
Email: 241501234@rajalakshmi.edu.in
Roll no: 241501234
Phone: null
Branch: REC
Department: l AI & ML FC
Batch: 2028
Degree: B.E - AI & ML

NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 1_COD_Question 1

Attempt : 1
Total Mark : 10
Marks Obtained : 0

Section 1 : Coding

1. Problem Statement

Janani is a tech enthusiast who loves working with polynomials. She wants to create a program that can add polynomial coefficients and provide the sum of their coefficients.

The polynomials will be represented as a linked list, where each node of the linked list contains a coefficient and an exponent. The polynomial is represented in the standard form with descending order of exponents.

Input Format

The first line of input consists of an integer n, representing the number of terms in the first polynomial.

The following n lines of input consist of two integers each: the coefficient and the exponent of the term in the first polynomial.

The next line of input consists of an integer m, representing the number of terms in the second polynomial.

The following m lines of input consist of two integers each: the coefficient and the exponent of the term in the second polynomial.

Output Format

The output prints the sum of the coefficients of the polynomials.

Sample Test Case

Input: 3
2 2
3 1
4 0
3
2 2
3 1
4 0
Output: 18

Answer

-

Status : Skipped   *Marks : 0/10*

Rajalakshmi Engineering College

Name: Udhayan Shejay VS
Email: 241501234@rajalakshmi.edu.in
Roll no: 241501234
Phone: null
Branch: REC
Department: l AI & ML FC
Batch: 2028
Degree: B.E - AI & ML

NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 1_COD_Question 2

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Arun is learning about data structures and algorithms. He needs your help in solving a specific problem related to a singly linked list.

Your task is to implement a program to delete a node at a given position. If the position is valid, the program should perform the deletion; otherwise, it should display an appropriate message.

Input Format

The first line of input consists of an integer N, representing the number of elements in the linked list.

The second line consists of N space-separated elements of the linked list.

The third line consists of an integer x, representing the position to delete.

Position starts from 1.

Output Format

The output prints space-separated integers, representing the updated linked list after deleting the element at the given position.

If the position is not valid, print "Invalid position. Deletion not possible."

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
8 2 3 1 7
2
Output: 8 3 1 7

Answer

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

void insert(int);
void display_List();
void deleteNode(int);

struct node {
    int data;
    struct node* next;
} *head = NULL, *tail = NULL;

// You are using GCC
void deleteNode(int pos){
    if(pos<=0){
        printf("Invalid position.Deletion not possible.");
        return;
    }

    struct node*temp=head;
    struct node*prev=NULL;
    int i;
```

```c
    for(i =1;i<pos &&  temp != NULL;i++){
        prev = temp;
        temp = temp->next;
    }

    if(temp == NULL){
        printf("Invalid position.Deletion not possible.");
        return;
    }

    if(prev == NULL){
        head = head->next;
        free(temp);
    }else{
        prev->next=temp->next;
        free(temp);
    }

    display_List();
    return;
}

void insert(int value){
    struct node*newnode;
    newnode = (struct node*)malloc(sizeof(struct node));
    newnode->data = value;
    newnode->next = NULL;

    if(head==NULL){
    head=newnode;
    tail=newnode;

    }else{
        tail->next=newnode;
        tail = newnode;
    }

    return;
}
```

```c
void display_List(){
    struct node*temp;
    temp = head;
    while(temp!=NULL){
      if(temp->next==NULL){
        printf("%d",temp->data);
       }else{
         printf("%d",temp->data);
       }
       temp = temp->next;
    }
    return;
}


int main() {
   int num_elements, element, pos_to_delete;

   scanf("%d", &num_elements);

   for (int i = 0; i < num_elements; i++) {
       scanf("%d", &element);
       insert(element);
   }

   scanf("%d", &pos_to_delete);

   deleteNode(pos_to_delete);

   return 0;
}
```

Status : Correct   *Marks : 10/10*

Rajalakshmi Engineering College

Name: Udhayan Shejay VS
Email: 241501234@rajalakshmi.edu.in
Roll no: 241501234
Phone: null
Branch: REC
Department: l AI & ML FC
Batch: 2028
Degree: B.E - AI & ML

NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 1_COD_Question 3

Attempt : 1
Total Mark : 10
Marks Obtained : 0

Section 1 : Coding

1. Problem Statement

Imagine you are working on a text processing tool and need to implement a feature that allows users to insert characters at a specific position.

Implement a program that takes user inputs to create a singly linked list of characters and inserts a new character after a given index in the list.

Input Format

The first line of input consists of an integer N, representing the number of characters in the linked list.

The second line consists of a sequence of N characters, representing the linked list.

The third line consists of an integer index, representing the index(0-based) after

which the new character node needs to be inserted.

The fourth line consists of a character value representing the character to be inserted after the given index.

Output Format

If the provided index is out of bounds (larger than the list size):

1. The first line of output prints "Invalid index".
2. The second line prints "Updated list: " followed by the unchanged linked list values.

Otherwise, the output prints "Updated list: " followed by the updated linked list after inserting the new character after the given index.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
a b c d e
2
X
Output: Updated list: a b c X d e

Answer

-

Status : Skipped   **Marks : 0/10**