Rajalakshmi Engineering College

Name: NETHRA CHANDRAGANDHI T
Email: 240701357@rajalakshmi.edu.in
Roll no: 240701357
Phone: 9487531086
Branch: REC
Department: I CSE FD
Batch: 2028
Degree: B.E - CSE

Scan to verify results

NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 4_CY

Attempt : 1
Total Mark : 30
Marks Obtained : 30

Section 1 : Coding

1. Problem Statement

Saran is developing a simulation for a theme park where people wait in a queue for a popular ride.

Each person has a unique ticket number, and he needs to manage the queue using a linked list implementation.

Your task is to write a program for Saran that reads the number of people in the queue and their respective ticket numbers, enqueue them, and then calculate the sum of all ticket numbers to determine the total ticket value present in the queue.

Input Format

The first line of input consists of an integer N, representing the number of people

in the queue.

The second line consists of N space-separated integers, representing the ticket numbers.

Output Format

The output prints an integer representing the sum of all ticket numbers.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
2 4 6 7 5
Output: 24

Answer

```
// You are using GCC
#include<stdio.h>
#define max 100
int q[max],f=-1,r=-1;
void eq(int d){
   if(f==-1)
   f=0;
   r++;
   q[r]=d;
}

void sot(){
   int s=0;
   for(int i=f;i<=r;i++){
      s+=q[i];
   }
   printf("%d",s);
}

int main(){
   int n,val,sum=0;
   scanf("%d",&n);
   for(int i=0;i<n;i++){
```

```
        scanf("%d",&val);
        eq(val);
    }
    sot();
}
```

Status : Correct **Marks : 10/10**

2. Problem Statement

A customer support system is designed to handle incoming requests using a queue. Implement a linked list-based queue where each request is represented by an integer. After processing the requests, remove any duplicate requests to ensure that each request is unique and print the remaining requests.

Input Format

The first line of input consists of an integer N, representing the number of requests to be enqueued.

The second line consists of N space-separated integers, each representing a request.

Output Format

The output prints space-separated integers after removing the duplicate requests.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
2 4 2 7 5

Output: 2 4 7 5

Answer

```
// You are using GCC
#include<stdio.h>
```

```
#define max 100
int q[max],f=-1,r=-1;
void eq(int d){
   if(f==-1)
   f=0;
   r++;
   q[r]=d;
}

void rd(int n){
   for(int i=0;i<n;i++){
      for(int j=i+1;j<n;){
         if(q[i]==q[j]){
            for(int k=j;k<n-1;k++){
               q[k]=q[k+1];
            }
            n--;
         }
         else{
            j++;
         }
      }
   }
   for(int i=0;i<n;i++){
      printf("%d ",q[i]);
   }
}

int main(){
   int n,val;
   scanf("%d",&n);
   for(int i=0;i<n;i++){
      scanf("%d",&val);
      eq(val);
   }
   rd(n);
}
```

Status : Correct **Marks : 10/10**

3. Problem Statement

Pathirana is a medical lab specialist who is responsible for managing blood count data for a group of patients. The lab uses a queue-based system to track the blood cell count of each patient. The queue structure helps in processing the data in a first-in-first-out (FIFO) manner.

However, Pathirana needs to remove the blood cell count that is positive even numbers from the queue using array implementation of queue, as they are not relevant to the specific analysis he is performing. The remaining data will then be used for further medical evaluations and reporting.

Input Format

The first line consists of an integer n, representing the number of a patient's blood cell count.

The second line consists of n space-separated integers, representing a blood cell count value.

Output Format

The output displays space-separated integers, representing the remaining blood cell count after removing the positive even numbers.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
1 2 3 4 5

Output: 1 3 5

Answer

```
#include<stdio.h>
#define max 100
int q[max],f=-1,r=-1;
void eq(int d){
   if(f==-1)
   f=0;
   r++;
```

```
    q[r]=d;
}

void bo(){
    for(int i=f;i<=r;i++){
        if(q[i]<0)
        printf("%d ",q[i]);
        else if(q[i]%2!=0)
        printf("%d ",q[i]);
    }
}

int main(){
    int n,val;
    scanf("%d",&n);
    for(int i=0;i<n;i++){
        scanf("%d",&val);
        eq(val);
    }
    bo();
}
```

Status : Correct **Marks : 10/10**