B23CS005

Q} Implement Circular Queue

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

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

struct Node *front = NULL;
struct Node *rear = NULL;

void enqueue(int d){
    struct Node *newNode;
    newNode = (struct Node *)malloc(sizeof(struct Node));
    newNode->data = d;
    newNode->next = NULL;
    if((rear == NULL) && (front == NULL)){
       front = rear = newNode;
       rear->next = front;
    }
    else{
        rear->next = newNode;
        rear = newNode;
        rear->next = front;
    }
}

void dequeue(){
    struct Node *temp;
    temp = front;

    if(front == NULL && rear == NULL){
        printf("Circular Queue is Empty!\n");
    }
    else if(front == rear){
        front = NULL;
        rear = NULL;
        free(temp);
    }
```

```c
    else{
        front = front->next;
        rear->next = front;
        free(temp);
    }
}

void display(){
    struct Node *temp;
    temp = front;
    if(front == NULL && rear == NULL){
        printf("Circular Queue is Empty!\n");
        return;
    }
    do{
        printf("%d ", temp->data);
        temp = temp->next;
    }while(temp != front);
    printf("\n");

}

int main(){
    int choice;
    do{
        int opt;
        printf("Enter 1 To Enqueue\n\t 2 To Dequeue\n\t");
        switch(opt){
            case 1: int data;
                    printf("Enter the number you want to add : ");
                    scanf("%d", &data);
                    enqueue(data);
                    printf("\n");
                    display();
                    break;

            case 2: dequeue();
                    printf("\n");
                    display();
                    break;

            default:printf("Wrong choice entered!\n");
        }
        printf("Enter 1 To Continue and 0 To exit :");
        scanf("%d", &choice);
```

```
    }while(choice == 1);
}
```

2}WAP to find even maximum in an Array

```
#include <bits/stdc++.h>

using namespace std;

int evenMaximum(vector<int> &arr){
    int max = INT_MIN;
    for(int i = 0; i < arr.size(); i++){
        if(arr[i]%2 == 0 && max < arr[i]){
            max = arr[i];
        }
    }

    return max;
}

int main(){
    vector<int> arr;int len;
    cin >> len;
    for(int i = 0; i < len; i++){
        int element;
        cin >> element;
        arr.push_back(element);
    }

    int max = evenMaximum(arr);
    if(max == INT_MIN){
        cout << "No even Number in the array.";
    }
    else{
        cout << "Largest Even Number : "<< evenMaximum(arr);
    }

    return 0;
}
```

3}WAP to find maximum at ith position in an array

```
#include <bits/stdc++.h>
using namespace std;
```

```cpp
int MaximumAtPos(vector<int> &arr, int pos){
    int max = INT_MIN;
    for(int i = 0; i < pos+1; i++){
        if(max < arr[i])
           max = arr[i];
    }

    return max;
}

int main(){
    vector<int> arr;int len;
    int pos;
    cin >> len;
    cin >> pos;
    for(int i = 0; i < len; i++){
        int element;
        cin >> element;
        arr.push_back(element);
    }

    if(pos < 0 || pos >= len){
        cout << "Improper Position Entered!";
    }
    else{
        int max = MaximumAtPos(arr, pos);
        if(max == INT_MIN){
        cout << "No even Number in the array.";
         }
         else{
        cout << "Largest Even Number : " << max;

        }
    }
    return 0;
}
```