

Q. My Pair

Pair is used to combine together two values which may be different in type. Pair provides a way to store two heterogeneous objects as a single unit.

Create a pair of given two different type of values (int,string) and print them.

Mandatory

1. Create a pair named "mypair".

pair mypair;

2. Print the first and second value.

Refer Sample TestCases.

Programming Language need to be used:C++

Source Code

```
#include <iostream>
#include<utility>
using namespace std;
int main()
{
    pair<int,string>mypair;
    int a;
    string b;
    cin>>a>>b;
    mypair.first=a;
    mypair.second=b;
    cout<<mypair.first<<" ";
    cout<<mypair.second<<endl;
    return 0;
}
```

Sample Input

```
10
Bogar
```

Sample Output

```
10 Bogar
```

Result

Thus, Program " **My Pair** " has been successfully executed

Q. Remove Duplicate

Manasvi the technical head of the training centre in Chennai has planned to conduct the surprise test for his students.

She has given the task of removing the duplicate elements from a sorted linked list.

But she has imposed the following restriction to the students.

Mandatory:

Should use "push_back" function and "unique" function of "list" library of Standard template library.

Programming Language need to be used:C++

Source Code

```
#include <iostream>
#include <list>
using namespace std;
int main()
{
    list<int>m;
    int n;
    cin>>n;
    int x;
    for(int i=0;i<n;i++)
    {
        cin>>x;
        m.push_back(x);
    }
    m.unique();
    for (auto v:m)
        cout<<v<<" ";
    return 0;
}
```

Sample Input

```
15
5 5 5 5 7 9 9 10 11 11 12 15 15 18 18
```

Sample Output

```
5 7 9 10 11 12 15 18
```

Result

Thus, Program " **Remove Duplicate** " has been successfully executed

Q. Address Map

Naresh the programmer has given the task to his juniors in the team.

The task is creating a map of type and add given Keys and Values.

Erase the Entry with given key and print the Map.

Mandatory:-

1. Create a map mymap using map mymap;
2. Insert values to map using .insert()
3. Erase the Entry using .erase(key_to_deleted);
4. Traverse and print the values using iterator.

Refer sample testcases.

Programming Language need to be used:C++

Source Code

```
#include <iostream>
#include<map>
using namespace std;
int main()
{
    int n,key_to_deleted=3,a[100],b[100],i,j;
    cin>>n;
    map<int,int>mymap;
    for(i=0;i<n;i++)
    {
        cin>>a[i]>>b[i];
    }
    for(j=0;j<n;j++)
        mymap.insert((a[j],b[j]));
    cin>>key_to_deleted;
    mymap.erase(key_to_deleted);
    map<int,int>::iterator it1;
    for(it1=mymap.begin();it1!=mymap.end();++it1)
        cout<<it1->first<<" "<<it1->second<<endl;
    return 0;
}
```

Sample Input

```
5
1 1
2 12
3 123
4 1234
5 12345
3
```

Sample Output

```
1 1
2 12
4 1234
5 12345
```

Result

Thus, Program " **Address Map** " has been successfully executed

Q. Sort Game

You are given N integers. Sort the N integers and print the sorted order.

Store the N integers in a vector.

Vectors are sequence containers representing arrays that can change in size.

Declaration:

vector<int> v; (creates an empty vector of integers)

Size:

int size=v.size();

Pushing an integer into a vector:

v.push_back(x); (where x is an integer. The size increases by 1 after this.)

Popping the last element from the vector:

v.pop_back(); (After this the size decreases by 1)

Sorting a vector:

sort(v.begin(), v.end()); (Will sort all the elements in the vector)

Input Format

The first line of the input contains N where N is the number of integers.

The next line contains N integers .

Constraints

1 ≤ N ≤ 10⁵

1 ≤ V_i ≤ 10⁹

where V_i is the ith integer in the vector.

Output Format

Print the integers in the sorted order one by one in a single line followed by a space.

Source Code

```
#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin>>n;
    int x;
    vector<int> v;
    while(n-->0)
    {
        cin>>x;
        v.push_back(x);
    }
    sort(v.begin(), v.end());
    for(auto i:v)
    {
        cout<<i<<" ";
    }
    return 0;
}
```

Sample Input

```
5
1 6 10 8 4
```

Sample Output

```
1 4 6 8 10
```

Result

Thus, Program " **Sort Game** " has been successfully executed

Q. Swapping two Functions

Vidhya the professor of SRM University has planned to conduct a surprise test for her students. The task assigned to the students is to create a swap function which swap two stacks and print the final result. She has imposed some of the restrictions in completing the task as follows.

Mandatory:

1. Should use "stack" library and "push" , "pop" functions of Standard template Library.
2. Create 2 vectors named "i" and "j" to complete the task
3. Use reverse function of STL library

Refer Sample testcases.

Programming Language need to be used: C++

Source Code

```
#include <stack>
#include <iostream>
#include <vector>
#include <bits/stdc++.h>
using namespace std;
int main()
{
    stack<int> mystack1;
    stack<int> mystack2;
    vector<int> i;
    vector<int> j;
    int n,k,a;
    cin>>n;
    for(k=0;k<n;k++)
    {
        cin>>a;
        mystack1.push(a);
    }
    for(k=0;k<n;k++)
    {
        cin>>a;
        mystack2.push(a);
    }
    mystack1.swap(mystack2);
    cout<<" ";
    while(!mystack1.empty())
    {
        cout<<mystack1.top()<<" ";
        mystack1.pop();
    }
    reverse(i.begin(),i.end());
    reverse(j.begin(),j.end());
    cout<<endl;
    while(!mystack2.empty())
    {
        cout<<mystack2.top()<<" ";
        mystack2.pop();
    }
    return 0;
}
```

Sample Input

```
4
1 2 3 4
5 6 7 8
```

Sample Output

```
8 7 6 5
4 3 2 1
```

Result

Thus, Program " **Swapping two Functions** " has been successfully executed

Q. Play with Streams

stringstream is a stream class to operate on strings. It basically implements input/output operations on memory (string) based streams.

stringstream can be helpful in different type of parsing.

The following operators/functions are commonly used here

1.Operator >>

Extracts formatted data.

2.Operator <<

Inserts formatted data.

3.Method str()

Gets the contents of underlying string device object.

4.Method str(string)

Sets the contents of underlying string device object.

Mandatory:

1.You have to write the function vector parseInts(string str)

2.str will be a string consisting of comma-separated integers, and you have to return a vector of int representing the integers.

Note:Header files need to be included without any spaces.

Input Format

The first and only line consists of n integers separated by commas.

Output Format

Print the integers after parsing it.

Refer Sample test cases.

Programming Language need to be used:C++

Source Code

```
#include<sstream>
#include<vector>
#include <iostream>
using namespace std;
vector<int>parseInts(string str)
{
    stringstream ss(str);
    vector<int>result;
    int temp_int;
    char temp_char;
    ss>>temp_int;
    result.push_back(temp_int);
    while(ss>>temp_char)
    {
        ss>>temp_int;
        result.push_back(temp_int);
    }
    return result;
}
int main()
{
    string str;
    cin>>str;
    vector<int>integers=parseInts(str);
    for(int i=0;i<integers.size();i++)
    {
        cout<<integers[i]<<"\n";
    }
    return 0;
}
```

Sample Input

23,4,56

Sample Output

23
4
56

Result

Thus, Program " **Play with Streams** " has been successfully executed

Q. Deque

Given a set of arrays of size N and an integer K, you have to find the maximum integer for each and every contiguous subarray of size K for each of the given arrays.

Input Format

First line of input will contain the number of test cases T.

For each test case, you will be given the size of array N and the size of subarray to be used K.

This will be followed by the elements of the array Ai.

Output Format

For each of the contiguous subarrays of size K of each array, you have to print the maximum integer.

Mandatory:

1. Should Use "deque" class
2. Use "push_back" and "pop_back" function of deque class

Refer Sample Test Cases.

Programming Language need to be used:C++

Source Code

```
#include<cstdio>
#include<deque>
#include<algorithm>
#include <iostream>
using namespace std;
int a[1000000];
int x[1000000],y[1000000];
deque<int>dq2;
int main()
{
    int T;
    cin>>T;
    while(T-->0)
    {
        dq2.clear();
        int n,k;
        scanf("%d %d",&n,&k);
        for(int i=0;i<n;i++)
            scanf("%d",&a[i]);
        for(int i=0;i<k-1;i++)
        {
            while(dq2.size()>0&&a[dq2[dq2.size()-1]]<=a[i])
                dq2.pop_back();
            dq2.push_back(a[i]);
        }
        for(int i=0,j=i+k-1;j<n;j++)
        {
            while(dq2.size()>0&&a[dq2[dq2.size()-1]]<=a[j])
                dq2.pop_back();
            dq2.push_back(a[j]);
            y[i]=a[dq2[0]];
            if(dq2[0]==a[i])
                dq2.pop_front();
        }
        for(int i=0;i<n-k;i++)
            printf("%d%c",y[i],i==n-k?"\n":' ');
    }
    return 0;
}
```

Sample Input

```
2
5 2
3 4 6 3 4
7 4
3 4 5 8 1 4 10
```

Sample Output

```
4 6 6 4
8 8 8 10
```

Result

Thus, Program " **Deque** " has been successfully executed

Q. Marks and Vector

You are appointed as the assistant to a teacher in a school and she is correcting the answer sheets of the students.

Your task is to calculate the marks given by the teacher and to Store the given values into a vector and find the maximum and minimum value using *min_element and *max_element.

Mandatory:-

1. Create vector using vector myvector;
2. Find maximum and minimum value using *min_element and *max_element.

Refer Sample testcases.

Programming Language need to be used:C++

Source Code

```
#include <iostream>
#include <algorithm>
#include<vector>
using namespace std;
vector<int> myvector;
int main()
{
    int n;
    cin>>n;
    int num=0;
    for(int i=0;i<n;i++)
    {
        cin>>num;
        myvector.push_back(num);
    }
    cout<<*min_element(myvector.begin(),myvector.end())<<" ";
    cout<<*max_element(myvector.begin(),myvector.end());
    return 0;
}
```

Sample Input

```
5
1 6 5 5 1
```

Sample Output

```
1 6
```

Result

Thus, Program " **Marks and Vector** " has been successfully executed

Q. Sets

Sets are containers that store unique elements following a specific order.

HINT:

Here are some of the frequently used member functions of sets:

sets; //Creates a set of integers.

int length=s.size(); //Gives the size of the set.

s.insert(x); //Inserts an integer x into the set s.

s.erase(val); //Erases an integer val from the set s.

Coming to the problem, you will be given Q queries. Each query is of one of the following three types:

1 x: Add an element x to the set.

2 x: Delete an element x from the set. (If the number x is not present in the set, then do nothing).

3 x: If the number x is present in the set, then print "Yes" (without quotes) else print "No" (without quotes).

Input Format

The first line of the input contains Q where Q is the number of queries.

The next Q lines contain 1 query each.

Each query consists of two integers y and x where y is the type of the query and x is an integer.

Constraints

1<=Q<=10 power 5

1<=y<=3

1<=x<=10 power 9

Output Format

For queries of type 3 print "Yes"(without quotes) if the number x is present in the set and if the number is not present, then print "No"(without quotes).

Each query of type 3 should be printed in a new line.

Source Code

```
#include<cmath>
#include<cstdio>
#include<vector>
#include<iostream>
#include<algorithm>
#include<set>
using namespace std;
int main()
{
    int n;
    set<int>s;
    cin>>n;
    while(n-->0)
    {
        int x,y;
        cin>>y>>x;
        if(y==1)
        {
            s.insert(x);
        }
        else if(y==2)
        {
            s.erase(x);
        }
        else
        {
            auto itr=s.find(x);
            if(distance(itr,s.end())==0)
            {
                cout<<"No"<<endl;
            }
            else
            {
                cout<<"Yes"<<endl;
            }
        }
    }
    return 0;
}
```

Sample Input

```
8
1 9
1 6
1 10
1 4
3 6
3 14
2 6
3 6
```

Sample Output

```
Yes
No
No
```

Result

Thus, Program " **Sets** " has been successfully executed

Q. Vector Iterator

Your task is to Create a vector and to add the given values to it.

Then you need to print the vector values in given order and reverse order using iterator and reverse iterator.

Mandatory:

1. Create a vector named "MyVector".
2. Create a iterator and reverse iterator.
3. Add the values into vector using push_back() function.
4. Use iterator and reverse_iterator to traverse and print the values

Refer Sample Test cases.

Programming Language need to be used:C++

Source Code

```
#include<iostream>
#include<vector>
#include<iterator>
using namespace std;
int main()
{
    int num,n;
    cin>>n;
    vector<int>MyVector;
    for(int i=0;i<n;i++)
    {
        cin>>num;
        MyVector.push_back(num);
    }
    vector<int>::iterator ptr;
    for(ptr=MyVector.begin();ptr<MyVector.end();ptr++)
        cout<<"ptr<<" ";
    cout<<endl;
    vector<int>::reverse_iterator ptr1;
    for(ptr1=MyVector.rbegin();ptr1<MyVector.rend();ptr1++)
        cout<<"ptr1<<" ";
    return 0;
}
```

Sample Input

```
3
17 56 34
```

Sample Output

```
17 56 34
34 56 17
```

Result

Thus, Program " **Vector Iterator** " has been successfully executed