**Problem No: 01**

**Topic: The Foundations: Logic and Proof**

**Problem Title:**

The 3x+1 Conjecture: Let T be the transformation that sends an even integer x to x/2 and an odd integer x to 3x + 1. A famous conjecture, sometimes known as the 3x + 1 conjecture, states that for all positive integers x, when we repeatedly apply the transformation T, we will eventually reach the integer 1.

**Source Code:**

#include <iostream>

using namespace std;

int e(int n);

int o(int n);

int main()

{

int n, n0;

cout << "Enter an integer: ";

cin >> n0;

n = n0;

while(1){

if(n % 2 == 0){

cout << "T(" << n << ") = " << e(n) << endl;

n = e(n);

}

else{

cout << "T(" << n << ") = " << o(n) << endl;

n = o(n);

}

if(n == 1)

break;

}

return 0;

}

int e(int n){return (n / 2);}

int o(int n){return (3 \* n + 1);}

**Output:**

Enter an integer: 30

T(30) = 15

T(15) = 46

T(46) = 23

T(23) = 70

T(70) = 35

T(35) = 106

T(106) = 53

T(53) = 160

T(160) = 80

T(80) = 40

T(40) = 20

T(20) = 10

T(10) = 5

T(5) = 16

T(16) = 8

T(8) = 4

T(4) = 2

T(2) = 1

Process returned 0 (0x0) execution time : 1.378 s

Press any key to continue.

**Problem No: 02**

**Topic: Basic Structures: Sets**

**Problem Title:**

Given subsets A and B of a set with n elements, find A, A ∪ B, A ∩ B, A – B and A ⊕ B.

**Objectives:**

To learn set operations like Union, Intersection, Difference and Symmetric Difference.

**Theory:**

A ∪ B = {x | x ∈ A ∨ x ∈ B}

A ∪ B = {x | x ∈ A  x ∈ B}

A – B = {x | x ∈ A  x ∉ B} = A ∩B

A ⊕ B = (A – B) ∪ (B – A)

**Source Code:**

#include <bits/stdc++.h>

#include <vector>

using namespace std;

void sort(int \*a, int n);

int main()

{

int na, nb;

vector <int> u;

vector <int> is;

vector <int> d;

vector <int> e;

vector <int> s;

cout << "Enter no. of elements of set A: ";

cin >> na;

cout << "Enter no. of elements of set B: ";

cin >> nb;

int i, a[na], j, b[nb];

cout << "Enter elements of set A: ";

for(i = 0; i < na; i++)

cin >> a[i];

cout << "Enter elemenst of set B: ";

for(i = 0; i < nb; i++)

cin >> b[i];

sort(a, na);

sort(b, nb);

i = 0; j = 0;

while(i < na || j < nb){

if(a[i] < b[j]){

u.push\_back(a[i]);

i++;

}

else if(a[i] > b[j]){

u.push\_back(b[j]);

j++;

}

else if(a[i] == b[j]){

u.push\_back(a[i]);

i++; j++;

}

}

cout << "\nUnion of A and B: ";

for(i = 0; i < u.size(); i++)

cout << u[i] << " ";

cout << endl;

i = 0; j = 0;

while(i < na || j < nb){

if(a[i] < b[j])

i++;

else if(a[i] > b[j])

j++;

else if(a[i] == b[j]){

is.push\_back(a[i]);

i++; j++;

}

}

cout << "\nIntersection of A and B: ";

for(i = 0; i < is.size(); i++)

cout << is[i] << " ";

cout << endl;

i = 0; j = 0;

while(i < na || j < nb){

if(a[i] < b[j]){

d.push\_back(a[i]);

i++;

}

else if(a[i] > b[j])

j++;

else if(a[i] == b[j]){

i++; j++;

}

}

cout << "\nDifference of A and B: ";

for(i = 0; i < d.size(); i++)

cout << d[i] << " ";

cout << endl;

i = 0; j = 0;

while(i < na || j < nb){

if(a[i] > b[j]){

e.push\_back(b[j]);

j++;

}

else if(a[i] < b[j])

i++;

else if(a[i] == b[j]){

i++; j++;

}

}

cout << "\nDifference of B and A: ";

for(i = 0; i < e.size(); i++)

cout << e[i] << " ";

cout << endl;

i = 0; j = 0;

while(i < d.size() || j < e.size()){

if(d[i] < e[j]){

s.push\_back(d[i]);

i++;

}

else if(d[i] > e[j]){

s.push\_back(e[j]);

j++;

}

else if(d[i] == e[j]){

s.push\_back(d[i]);

i++; j++;

}

}

cout << "\nSymmetric Difference of A and B: ";

for(i = 0; i < s.size(); i++)

cout << s[i] << " ";

cout << endl;

return 0;

}

void sort(int \*a, int n)

{

int i, j, t;

for(i = 0; i < n - 1; i++){

for(j = 0; j < n - 1 - i; j++){

if(a[j] > a[j + 1]){

t = a[j];

a[j] = a[j + 1];

a[j + 1] = t;

}

}

}

}

**Output:**

Enter no. of elements of set A: 4

Enter no. of elements of set B: 4

Enter elements of set A: 1 2 3 4

Enter elemenst of set B: 3 4 5 6

Union of A and B: 1 2 3 4 5 6

Intersection of A and B: 3 4

Difference of A and B: 1 2

Difference of B and A: 5 6

Symmetric Difference of A and B: 1 2 5 6

Process returned 0 (0x0) execution time : 6.760 s

Press any key to continue.