TASK-3

#include<bits/stdc++.h>

using namespace std;

class Candidates

{

private:

static int count;

int exam\_id;

string phone;

int exam1,exam2;

int merit;

public:

set\_phone(string s)

{

phone = s;

}

string get\_phone()

{

return phone;

}

int get\_id()

{

return exam\_id;

}

int get\_total()

{

return count;

}

set\_exam1(int e1)

{

exam1 = e1;

}

set\_exam2(int e2)

{

exam2 = e2;

}

int get\_exam1()

{

return exam1;

}

int get\_exam2()

{

return exam2;

}

set\_merit(int n)

{

merit = n;

}

int get\_merit()

{

return merit;

}

void assign\_merit(Candidates[],int n)

{

int marks[n];

for(int i=0;i<n;i++)

{

marks[i] = get\_exam1() + get\_exam2();

}

for(int i=0;i<n;i++)

{

int pos = 1;

for(int j=0;j<n;j++)

{

if(j==i) continue;

else if(marks[j]>marks[i])

{

pos++;

}

}

Candidates[i].set\_merit(pos);

}

}

int highest\_mark(Candidates[],int n)

{

for(int i=0;i<n;i++)

{

if(Candidates[i].get\_merit() == 1)

{

return Candidates[i].get\_exam1() + Candidates[i].get\_exam2();

}

}

}

Candidates(int e1, int t\_e1, int e2, int t\_e2)

//t\_e1 and t\_e2 means total marks for exam1 and exam 2

{

count++;

exam\_id = count;

set\_phone("");

set\_exam1((e1\* 1000)/t\_e1);

set\_exam2((e2\*1000)/t\_e2);

}

};

int Candidates::count = 0;