

**CS-114 - Fundamental of Programing**

**Lab Manual # 10**

**Lab Task**

Submitted To: Engr Muhammad Affan

Submitted By: Shahzaib Murtaza (466034)

Section: B

**Task # 1**

#include <iostream>

#include <vector>

using namespace std;

int main(){

vector<int> a;

vector<int>::iterator rem=a.begin()+5;

for(int i=1; i<11; i++){

a.push\_back(i);

}

cout<<"The values in the vector are: \n";

for(vector<int>::iterator i=a.begin(); i!=a.end(); i++){

cout<<\*i<<'\t';

}

cout<<endl;

a.push\_back(5);

rem=a.begin()+5;

a.erase(rem);

cout<<"After pushing 5 and removing integer at posiiton 5"<<endl;

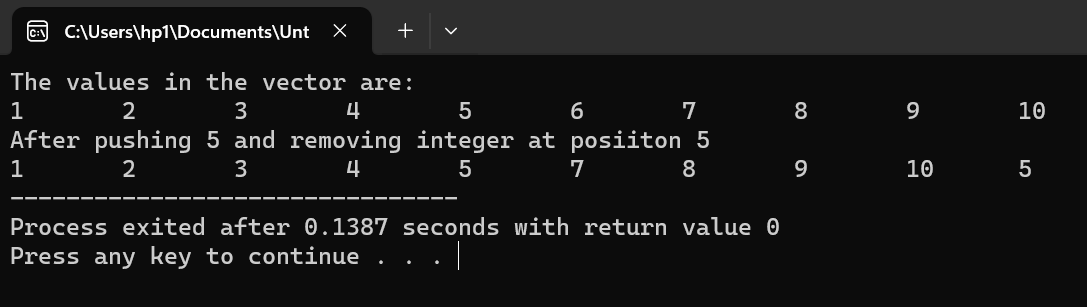
for(vector<int>::iterator i=a.begin(); i!=a.end(); i++){

cout<<\*i<<'\t';

}

return 0;

}



**Task # 2**

#include<iostream>

#include<vector>

using namespace std;

void sort(vector<int>& values) {

int n=values.size();

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

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

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

int temp=values[j];

values[j]=values[j+1];

values[j+1]=temp;

}

}

}

}

int median(vector<int>& values) {

sort(values);

int n=values.size();

int median;

if(n%2==0) {

median=(values[n/2-1]+values[n/2])/2;

}

else {

median=values[n/2];

}

return median;

}

int mode(const vector<int>& values) {

int maxfrequency=0;

int modeValue=0;

for(int i=0;i<values.size();i++) {

int frequency=0;

for(int j=0;j<values.size();j++) {

if(values[i]==values[j]) {

frequency++;

}

}

if(frequency>maxfrequency) {

maxfrequency=frequency;

modeValue=values[i];

}

}

return modeValue;

}

void equalgrades(const vector<string>& names,const vector<int>& grades,int mode) {

cout<<"Students with GradeEqual to Modevalue:";

for(int i=0;i<names.size();i++) {

if(grades[i]==mode) {

cout<<names[i]<<" ";

}

}

cout<<endl;

}

int main() {

vector<string> studentNames;

vector<int> studentGrades;

int num,mean,medianValue,modeValue;

cout<<"What is the number of students:";

cin>>num;

for(int i=0;i<num;i++) {

string name;

int grade;

cout<<"Name of student:";

cin>>name;

studentNames.push\_back(name);

cout<<"Enter the grade percentage:";

cin>>grade;

studentGrades.push\_back(grade);}

int sum=0;

for(int i=0;i<studentGrades.size();i++) {

sum+=studentGrades[i];}

mean=sum/studentGrades.size();

cout<<"Mean:"<<mean<<endl;

medianValue=median(studentGrades);

cout<<"Median:"<<medianValue<<endl;

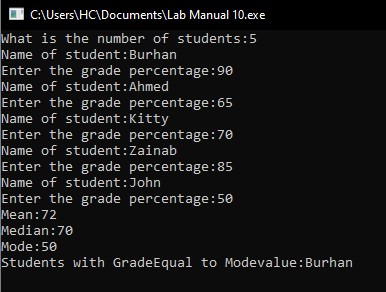
modeValue=mode(studentGrades);

cout<<"Mode:"<<modeValue<<endl;

equalgrades(studentNames,studentGrades,modeValue);

return 0;

}



**Task # 3**

#include <bits/stdc++.h>

using namespace std;

class Triangle{

public:

float dimension1, dimension2, dimension3;

float area(){

return (dimension1\*dimension2)/2;

}

float perimeter(){

return dimension1+dimension2+dimension3;

}

};

int main(){

Triangle test\_1;

cout<<"Input the dimesntions of the triangle. \n"

<<"Base: ";

cin>>test\_1.dimension1;

cout<<"Height: ";

cin>>test\_1.dimension2;

cout<<"Hypotenuse: ";

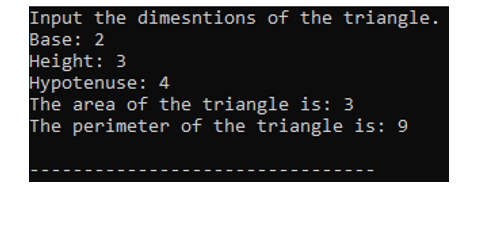
cin>>test\_1.dimension3;

cout<<"The area of the triangle is: "<<test\_1.area()<<endl

<<"The perimeter of the triangle is: "<<test\_1.perimeter()<<endl;

return 0;

}



**Task # 4**

#include <bits/stdc++.h>

using namespace std;

struct salary\_table{

string name;

int base\_pay, work\_hours, salary;

void pay\_inc(){

if(work\_hours<8){

salary=base\_pay;

}

else if(work\_hours<10 && work\_hours>=8){

salary=base\_pay+50;

}

else if(work\_hours<12 && work\_hours>=10){

salary=base\_pay+100;

}

else{

salary=base\_pay+150;

}

}

};

int main(){

int num, check;

cout<<"Input the number of employees: ";

cin>>num;

salary\_table employee[num];

for(int i=0; i<num; i++){

cout<<"Input the Name, Base Pay and Hours of Work Per Day of Employee "<<i+1<<endl;

cin>>employee[i].name;

cin>>employee[i].base\_pay;

cin>>employee[i].work\_hours;

employee[i].pay\_inc();

}

cout<<"Input the enrty number to check employe salary: \n"

<<"Input a charecter to exit \n";

do{

cin>>check;

if(check>=1 && check<=num){

cout<<employee[check-1].name<<'\t'

<<employee[check-1].work\_hours<<'\t'

<<employee[check-1].salary<<'\n';

}

else{

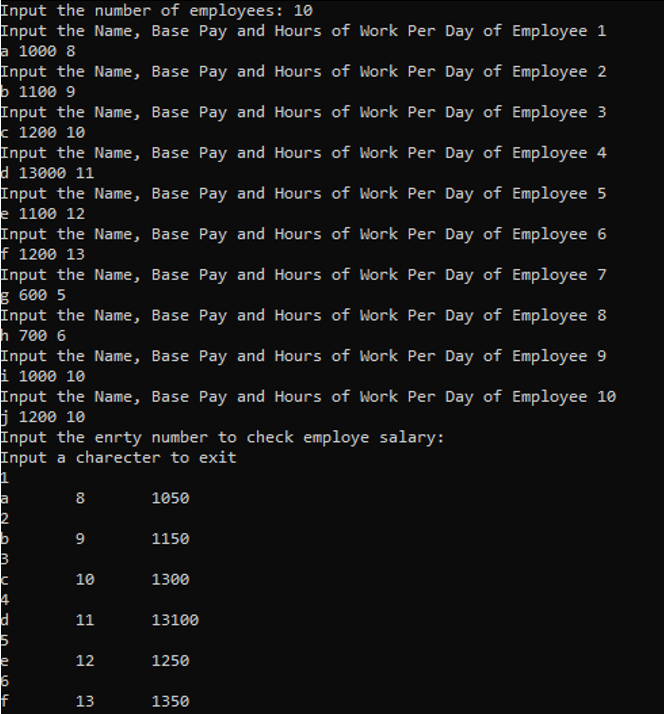
cout<<"Enrty number does not exist."<<endl;

}

} while(!cin.fail());

return 0;

}

****