**FESE-047 ANEEQ MURTAZA**

**Q2**

**A**

**#include <iostream>**

**using namespace std;**

**class Publications**

**{**

**};**

**class Staff**

**{**

**};**

**class Library**

**{**

**private:**

**Staff\* staff[20];**

**Publications\* publications[20];**

**string publishDate;**

**int staffId;**

**public:**

**Library() {**

**for(int i = 0; i < 20; i++) {**

**staff[i] = NULL;**

**}**

**for(int i = 0; i < 20; i++) {**

**publications[i] = NULL;**

**}**

**};**

**void addStaff(Staff \*stf, int staffId) {**

**if (staffId >= 0 && staffId <= 15) {**

**staff[staffId] = stf;**

**}**

**}**

**Staff\* getStaff(int staffId) {**

**if (staffId >=0 && staffId <=20) {**

**return staff[staffId];**

**}**

**else {**

**return NULL;**

**}**

**}**

**void setStaffId(int id) {**

**staffId = id;**

**}**

**int getStaffId() {**

**return staffId;**

**}**

**void addPublications(Publications \*pub, int pubNo)**

**{**

**if (pubNo >= 0 && pubNo <= 20) {**

**publications[pubNo] = pub;**

**}**

**}**

**Publications\* getPublications(int pubNo) {**

**if (pubNo >=0 && pubNo <=20) {**

**return publications[pubNo];**

**}**

**else {**

**return NULL;**

**}**

**}**

**void setPublishDate(string date) {**

**publishDate = date;**

**}**

**string getPublishDate() {**

**return publishDate;**

**}**

**};**

**int main()**

**{**

**cout << "name: " << endl;**

**cout << "rollno: " << endl << endl;**

**Staff s[20];**

**Publications p[20];**

**Library l1;**

**for (int i = 0; i <= 20; i ++)**

**{**

**l1.addStaff(&s[i], i);**

**cout << "Staff with staff id: " << i << " added"<< endl;**

**}**

**cout << endl;**

**for (int i = 0; i <= 20; i ++)**

**{**

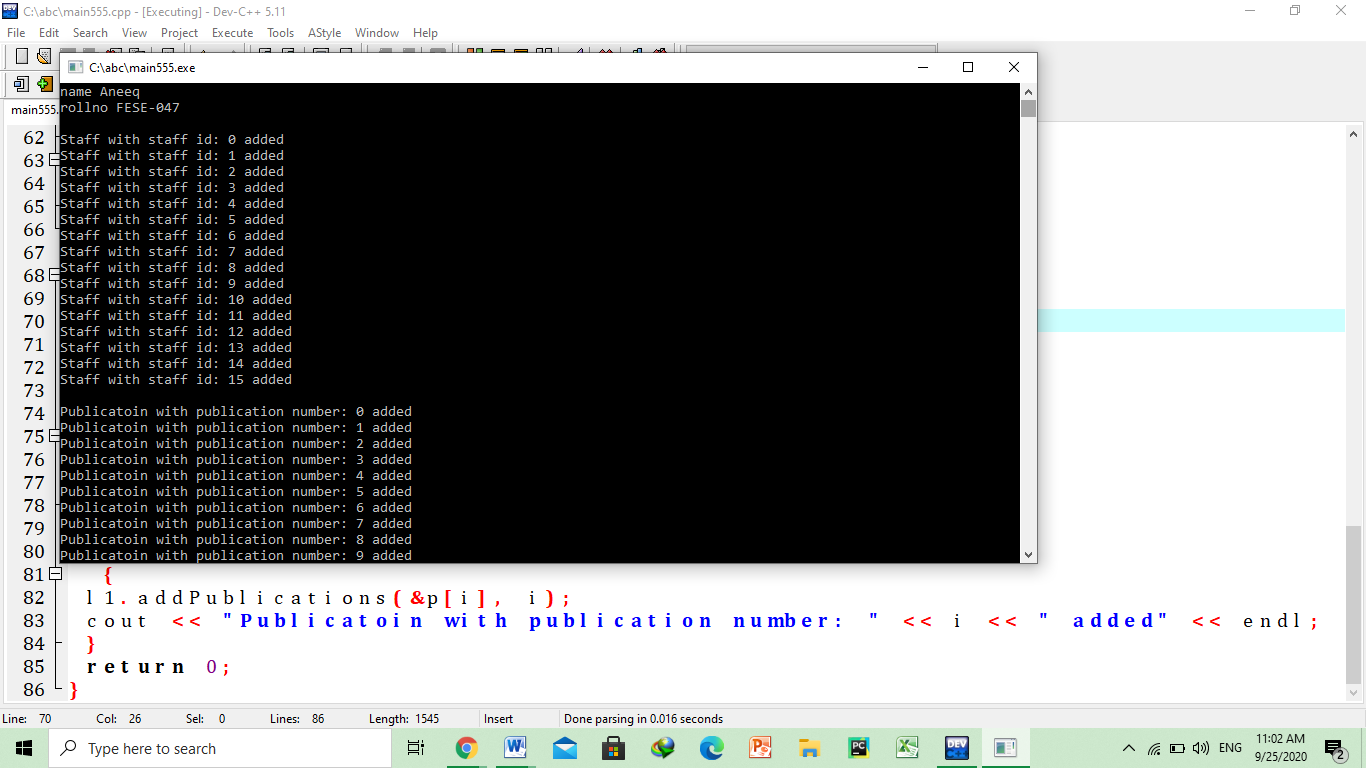
**l1.addPublications(&p[i], i);**

**cout << "Publicatoin with publication number: " << i << " added" << endl;**

**}**

**return 0;**

**}**



**B**

**#include <iostream>**

**#include <math.h>**

**#include <string.h>**

**#include <cmath>**

**using namespace std;**

**class triangle{**

**protected:**

**double base;**

**double altitude;**

**double angle;**

**public:**

**triangle(){};**

**triangle(double a,double ang){**

**base=a;**

**angle=ang;**

**}**

**triangle(double b,double a,double ang){**

**base=b;**

**altitude=a;**

**angle=a;**

**}**

**double getbase(){**

**return base;**

**}**

**void setbase(double b){**

**base=b;**

**}**

**double getaltitude(){**

**return altitude;**

**}**

**void setaltitue(double b){**

**altitude=b;**

**}**

**double getangle(){**

**return angle;**

**}**

**void setangle(double b){**

**angle=b;**

**}**

**double area()**

**{**

**return (base\*altitude)/2;**

**}**

**};**

**class isoceles:public triangle{**

**isoceles(int a,double ang):triangle(a,ang)**

**{**

**}**

**virtual double area()**

**{**

**return (base\*altitude)/2;**

**}**

**};**

**class rightangle:public triangle**

**{**

**public:**

**rightangle(int a,int b){**

**angle=90;**

**base=a;**

**altitude=b;**

**}**

**virtual double area(){**

**return base\*altitude/2;**

**}**

**};**

**class Equilateral:public triangle{**

**public:**

**Equilateral(int a){**

**base=a;altitude=a;**

**angle=60;**

**}**

**virtual double area()**

**{**

**return (sqrt(3)\*pow(base,2))/4;**

**}**

**double perimeter()**

**{**

**return 3\*base;**

**}**

**};**

**int main(){**

**cout<<"area and perimeter of equilateral triangle is";**

**Equilateral e1(23);**

**cout<<"area is "<<e1.area()<<endl;**

**cout<<"perimeter is "<<e1.perimeter();**

**return 0;**

**}**

