National University of Computer and Emerging Sciences



**Lab Exercise 10**

For

# Object Oriented Programming Lab

|  |  |
| --- | --- |
| Name | Muhammad Zain |
| Roll no | 19F-0228 |
| Course Instructor(s) | Dr. Danish |
| Lab Instructor(s) | Mr. Mughees Ismail |
| Semester | Spring 2020 |

**FAST School of Computing**

**Instructions:**

|  |  |
| --- | --- |
| 1. | Make a word document with the naming convention “SECTION\_ LAB#\_ROLLNO” and put all your source code and snapshots of its output in it. Make sure your word file is formatted properly. |
| 2. | Plagiarism is strictly prohibited. |
| 3. | Do not discuss solutions with one another. |

**Useful links**

## Question#1

# Source code:

#include<iostream>

#include<string>

using namespace std;

class Package {

protected:

string name;

string address;

string city;

string state;

int zip;

float weight; //(in ounces) and

double cost; //per ounce to ship the package.

public:

Package()

{

cout << "Enter the name" << endl;

getline(cin, name);

cout << "Enter your Address" << endl;

getline(cin, address);

cout << "Enter your City" << endl;

getline(cin, city);

cout << "Enter your State" << endl;

cin >> state;

cout << "Enter the zip code" << endl;

cin >> zip;

cout << "Enter the weight " << endl;

{cin >> weight;

while (weight<0)

{

cout << "Invalid input : Weight must be positive" << endl;

cout << "Input again" << endl;

cin >> weight;

}

}

cout << "Enter the Cost per ounce" << endl;

cin >> cost;

}

virtual double calculateCost()

{

cost = weight\*cost;

return cost;

}

};

class TwoDayPackage :Package {

protected:

double flat\_fee;// that the shipping company charges for two-day-delivery service.

public:

TwoDayPackage()

{

cout << "Enter the cost of Flat Fee" << endl;

cin >> flat\_fee;

}

double calculateCost()

{

Package::calculateCost();

cost += flat\_fee;

cout << "The Cost associated with this shpping package is " << cost;

}

};

class OvernightPackage :Package {

protected:

double additional\_fee;

public:

OvernightPackage()

{

cout << "Enter the additional amount of charges for overnight " << endl;

cin >> additional\_fee;

}

double calculateCost()

{

Package::calculateCost();

cost = cost + (additional\_fee\*weight);

cout << "The Cost associated with Overnight shpping package is " << cost;

}

};

int main()

{

int choice;

cout << "Press 1 for Two day Shipping Package" << endl;

cout << "Press 2 for Overnight shipping package" << endl;

cin >> choice;

if (choice == 1)

{

TwoDayPackage object;

object.calculateCost();

}

else if (choice == 2)

{

OvernightPackage obj;

obj.calculateCost();

}

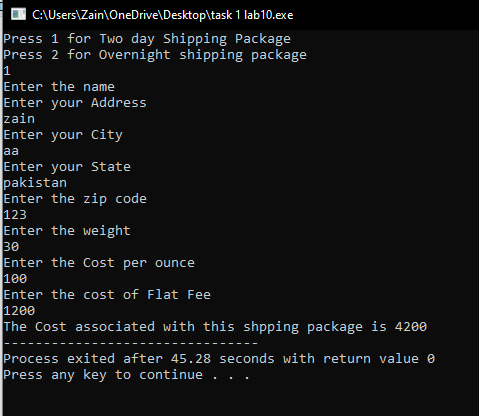
else

cout << "Invalid Input" << endl;

system("pause>0");

}

# Snip:



A screenshot of a cell phone on a table

Description automatically generated

## Question#2

#include<iostream>

#include<string>

#include<time.h>

#include<ctime>

#include<stdlib.h>

using namespace std;

class animal {

protected:

string name;

public:

animal()

{

name = "\_";

}

virtual void sound()

{

cout << "Virtual animal Sound Function" << endl;

}

};

class cat :public animal {

public:

cat()

{

name = "Cat";

}

void sound()

{

cout << "Cat Sound is Meow-Meow" << endl;

}

};

class dog :public animal {

public:

dog()

{

name = "Dog";

}

void sound()

{

cout << "Dog Barks" << endl;

}

};

class tiger :public animal {

public:

tiger()

{

name = "Tiger";

}

void sound()

{

cout << "Tiger Roars" << endl;

}

};

class deer :public animal {

public:

deer()

{

name = "deer";

}

void sound()

{

cout << "Deer bellows" << endl;

}

};

int main()

{

srand(time(0));

int size;

int match;

cout << "How many times you want to iterate the loop" << endl;

cin >> size;

animal \*\*obj\_animal;

obj\_animal = new animal \*[size];

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

{

match = rand() % 4 + 1;

if (match == 1)

{

cat obj\_cat;

obj\_animal[i] = &obj\_cat;

obj\_animal[i]->sound();

}

else if (match == 2)

{

dog obj\_dog;

obj\_animal[i] = &obj\_dog;

obj\_animal[i]->sound();

}

else if (match == 3)

{

tiger obj\_tiger;

obj\_animal[i] = &obj\_tiger;

obj\_animal[i]->sound();

}

else if (match == 4)

{

deer obj\_deer;

obj\_animal[i] = &obj\_deer;

obj\_animal[i]->sound();

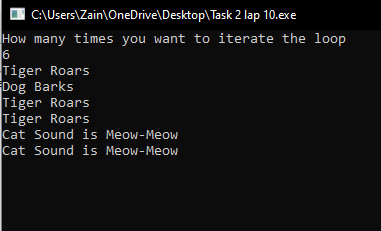
}

}

system("pause>0");

}

### **Snip:**

****

## }Question#3

# Source code:

#include<iostream>

#include<string>

#include<iomanip>

using namespace std;

class Employee {

protected:

string First\_Name;

string Last\_Name;

int ISSN;

public:

Employee()

{

First\_Name = "-";

Last\_Name = "-";

ISSN = 0;

}

void setFirstname()

{

cout << "Enter First Name: ";

cin >> First\_Name;

while (First\_Name == "")

{

cin >> First\_Name;

}

}

void getFirstname()

{

cout << First\_Name;

}

void setLastname()

{

cout << "Enter Last Name: ";

cin >> Last\_Name;

while (Last\_Name == "")

{

cin >> Last\_Name;

}

}

void getLastname()

{

cout << Last\_Name;

}

void setISSN() {

cout << "Enter the International Standard Serial Number (ISSN) " << endl;

cin >> ISSN;

while (ISSN == 0)

{

cin >> ISSN;

}

}

int getISSN()

{

return ISSN;

}

virtual void display()

{

cout << endl << "First Name : "; getFirstname();

cout << endl << "Last Name : "; getLastname();

cout << endl << "ISSN : "; getISSN();

}

};

class HourlyEmployee :public Employee {

protected:

float grossPay;

float netPay;

float tax;

float hoursWorked;

float rate;

public:

HourlyEmployee()

{

Employee::setFirstname();

Employee::setLastname();

Employee::setISSN();

cout << " Enter the Hours Worked" << endl;

cin >> hoursWorked;

cout << "Enter the Rate per Hour" << endl;

cin >> rate;

cout << " 5% of tax from your pay as per the Company Policy" << endl;

tax = 5;

}

void setGrosspay() {

grossPay = hoursWorked\*rate;

}

void getGrossPay()

{

cout << "Your Gross pay is " << grossPay << " $ " << endl;

}

void setNetpay()

{

netPay = (grossPay\*tax) / 100;

}

void getNetpay()

{

cout << "Your Net pay is " << netPay << " $ " << endl;

}

void display() {

Employee::display();

getGrossPay();

getNetpay();

}

};

class SalariedEmployee :virtual public Employee {

protected:

float MonthlygrossPay;

float MonthlynetPay;

float GpFund;

float daysWorked;

float earning;

public:

SalariedEmployee()

{

Employee::setFirstname();

Employee::setLastname();

Employee::setISSN();

cout << " Enter the Hours Worked" << endl;

cin >> daysWorked;

cout << "Enter the earning of One Day" << endl;

cin >> earning;

cout << " 10% GP fund is deducted" << endl;

GpFund = 10;

}

void setMonthlygrossPay()

{

MonthlygrossPay = daysWorked\*earning;

}

void getMonthlygrossPay()

{

cout << "Your Monthly Gross pay is " << MonthlygrossPay << " $ " << endl;

}

void setMonthlynetPay()

{

MonthlynetPay = (MonthlygrossPay\*GpFund) / 100;

}

void getMonthlynetPay()

{

cout << "Your Monthly pay is " << MonthlynetPay << " $ " << endl;

}

virtual void display() {

Employee::display();

getMonthlygrossPay();

getMonthlynetPay();

}

};

class Secratory :public Employee {

protected:

int sPass;

public:

Secratory() {

cout << "Thank you! Secratory for Entering your Data" << endl;

}

void spass()

{

cout << "Enter your Secratory Number" << endl;

cin >> sPass;

}

void getsPass()

{

cout << " Thank you! for entering the Secratory Number i.e " << sPass << endl;

}

void display()

{

Employee::display();

getsPass();

}

};

class Engineer :virtual public Employee {

protected:

int ePass;

public:

Engineer() {

cout << "Thank you! Engineer for Entering your Data" << endl;

}

void epass()

{

cout << "Enter your Engineer Number" << endl;

cin >> ePass;

}

void getePass()

{

cout << " Thank you! for entering the Engineer Number i.e " << ePass << endl;

}

virtual void display() {

Employee::display();

getePass();

}

};

class EngineeringManager : public SalariedEmployee, public Engineer

{

protected:

string Engineering;

public:

EngineeringManager()

{

cout << "Engineering Manager Welcome" << endl;

}

void setEngineering()

{

cout << "Please re-enter your Post" << endl;

cin >> Engineering;

}

void getPost()

{

cout << " your post is " << Engineering << endl;

}

void display()

{

Employee::display();

getPost();

}

};

int main()

{

int option;

cout << "Press 1 hourly employee class" << endl;

cout << "Press 2 for Salary Class" << endl;

cout << "Press 3 for Secratory Class" << endl;

cout << "Press 4 for Engineering Class" << endl;

cout << "Press 5 for Engineering manager Class" << endl;

cout << "Press 6 to exit" << endl;

cin >> option;

if (option == 1)

{

HourlyEmployee obj1;

obj1.setGrosspay();

obj1.setNetpay();

obj1.display();

}

else if (option == 2)

{

SalariedEmployee obj2;

obj2.setMonthlygrossPay();

obj2.setMonthlynetPay();

obj2.display();

}

else if (option == 3)

{

Secratory obj3;

obj3.spass();

obj3.display();

}

else if (option == 4)

{

Engineer obj4;

obj4.epass();

obj4.display();

}

else if (option == 5)

{

EngineeringManager obj5;

obj5.setEngineering();

obj5.display();

}

system("pause>0");

}

# Snip:

