Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования  
«Пермский национальный исследовательский политехнический университет»

(ПНИПУ)

**Факультет:** электротехнический

**Направление:** 09.03.04 Программная инженерия

ОТЧЁТ

По лабораторной работе №1

«Классы»

Семестр 2

Выполнил:

студент группы ПРТ-21-1б

Степванова И.В.

Пермь – 2022

**Задание №1**

**Постановка задачи**

Поле first – положительное целое число, номинал купюры; номенал может принимать значения 1, 2, 5, 10, 50, 100, 500, 1000, 5000, поле second – положительное целое число, колличество купюр данного достоинства. Реализовать метод summa() – вычисление денежной суммы.

**Анализ**

1. 1. Нужно разработать класс, содержащий два члена (first, second), и следующие методы:  
   – ввод с клавиатуры Read;  
   – вывод на экран Display;  
   – метод, указанный в задании.
2. Делаем 2 класса:  
   Первый - класс money содержит количество купюр данного номинала;  
   Второй - массив переменных money;
3. Полученный результат вывести => «summa".

**Используемые переменные**

int first – положительное целое число, номинал купюры; номинал может принимать значения 1, 2, 5. 10, 50, 100, 500, 1000, 5000;

int second – положительное целое число,  
количество купюр данного достоинства;

int summa – вычисление денежной суммы.

**Код C++**

#include "iostream"

#include <windows.h>

using namespace std;

class money

{

int first;

int second;

public:

int summa()

{

return first \* second;

}

void Display()

{

cout <<first<<':'<<second<<endl;

}

void Read(int nom);

friend class CashIn;

};

//------------------------------------------

class CashIn

{

static const int K=9;

int allowed[K];

money a[K];

public:

CashIn();

void ReadAll();

void DisplayAll();

int TotalSum();

};

//------------------------------------------

void money::Read(int nom)

{

first=nom;

cout <<"Enter amount:";

cin>>second;

}

//------------------------------------------

CashIn::CashIn() // конструктор

{

int t[K]={1,2,5,10,50,100,500,1000,5000};

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

allowed[j]=t[j];

}

void CashIn::ReadAll()

{

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

{

cout <<"Second "<<allowed[j]<<endl;

a[j].Read(allowed[j]);

}

}

void CashIn::DisplayAll()

{

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

a[i].Display();

}

int CashIn::TotalSum()

{

int totalsum=0;

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

totalsum+=a[i].summa();

return totalsum;

}

//------------------------------------------

int main() {

CashIn A;

A.ReadAll();

A.DisplayAll();

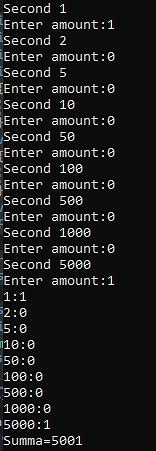
cout<<"Summa="<<A.TotalSum()<<endl;

system("pause");

return 0;

}

**Скриншоты результатов**



**Анализ результатов**

Пользователь вводит колличество купюр. Далее выводится сумма всех монет.

**Контрольные вопросы**

1. Что такое класс?

Класс – абстрактный тип данных, определяемый пользователем. Представляет собой модель реального объекта в виде данных и функций.

1. Что такое объект (экземпляр) класса?

Объект – переменная класса.

1. Как называются поля класса?

Полями класса называют данные, содержащиеся в классе.

1. Как называются функции класса?

Функциями класса называются методы.

1. Для чего используются спецификаторы доступа?

Спецификаторы доступа используются для изменения видимости некоторых компонентов класса.

1. Для чего используется спецификатор public?

Public используется для того, чтобы компоненты класса были открыты к доступу извне.

1. Для чего используется спецификатор private?

Private используется для того, чтобы компоненты класса были закрыты от доступа извне.

1. Если описание класса начинается со спецификатора class, то какой спецификатор доступа будет использоваться по умолчанию?

Private.

1. Если описание класса начинается со спецификатора struct, то какой спецификатор доступа будет использоваться по умолчанию?

Public.

1. Какой спецификатор доступа должен использоваться при описании интерфейса класса? Почему?

Должен использоваться спецификатор public для того, чтобы к методу можно было обратиться извне.

1. Каким образом можно изменить значения атрибутов экземпляра класса?

Изменить значение можно через методы, описанные в public, или, если поля публичные, обратиться к ним напрямую.

1. Каким образом можно получить значения атрибутов экземпляра класса?

Получить значение можно через методы, описанные в public, или, если поля публичные, обратиться к ним напрямую.

1. Класс описан следующим образом:

Struct Student

{

String name;

Int group;

…

};

Объект класса определен следующим образом

Student \*s = new Student;

Как можно обратиться к полю name объекта s?

S[i].name = “”;// i – номер элемента



Struct Student

{

String name;

Int group;

…

};

Student s;

Как можно обратиться к полю name объекта s?

s.name = “”;

15. class Student

{

String name;

Int group;

…

};

Student \*s = new Student;

Можно обратиться только через методы класса.

16.

class Student

{

String name;

Int group;

…

};

Объект класса определен следующим образом

Student s;

Как можно обратиться к полю name объекта s?

Можно обратиться только через методы класса.

17. Класс описан следующим образом:

class Student

{

Public:

String name;

Int group;

…

};

Объект класса определен следующим образом:

Student\* s = new Student;

Как можно обратиться к полю name объекта s?

S[i].name = “”; // i – номер элемента