**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ**

**КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

**КЫРГЫЗСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. И. Раззакова**

**ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

Кафедра: **Программное обеспечение компьютерных систем**

Дисциплина «**Объектно-Ориентированное Программирование**»

**Отчет**

**Лабораторная работа №1**

Выполнил: студент группы ПИ-2-21

Газиев Давид

Проверил: Мусабаев Э. Б.

**Бишкек 2024**

**Задание 1**

Возведение числа **n** в степень **p** – это умножение числа **n** на себя **p** раз.

Напишите функцию (не библиотечную) с именем **power** (), которая в качестве аргументов принимает значение типа **double** для **n** и значение типа **int** для **p** и возвращает значение типа **double**.

Для аргумента, соответствующего степени числа, задать значение по умолчанию, равное **2**, чтобы при отсутствии показателя степени при вызове функции число **n** возводилось в квадрат.

Сделайте проверку, будет ли пользователь вводить степень числа. Если будет, то эта степень – число, большее 2. Напишите функцию **main**(), которая запрашивает у пользователя ввод аргументов для функции **power** () и отображает на экране результаты её работы. Напишите код первого задания с *объявлением* (прототипом) функции power ().

**Код программы**

#include <iostream>

using namespace std;

double power(double n, int p = 2) {

double result = n;

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

result \*= n;

}

return result;

}

int main()

{

setlocale(LC\_ALL, "Russian");

float digit;

int powerDigit, answer;

double result;

cout << "Введите число: ";

cin >> digit;

cout << "Хотите ли ввести степень? (>=1 - да, 0 - нет): ";

cin >> answer;

if (answer) {

while (true) {

cout << "Введите степень: ";

cin >> powerDigit;

if (powerDigit > 2) break;

else cout << "Степень должна быть больше 2!\n";

}

result = power(digit, powerDigit);

}

else {

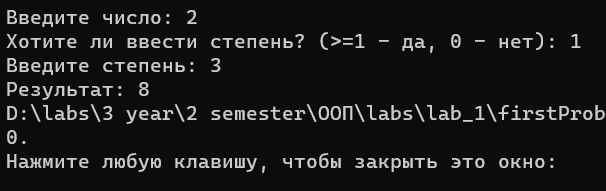
result = power(digit);

}

cout << "Результат: " << result;

}

**Проверка выполнения**

****

**Блок схема**



**Задание 2**

Взяв в качестве основы функцию **power** () из первого задания, работающую только со значением типа **double**. Создайте перегруженные функции с этим же именем, принимающими в качестве аргумента значения типа **char**, short **int**, **long int, float**. Напишите программу, вызывающую функцию **power** () со всеми возможными типами аргументов. Напишите код второго задания без *объявления* функции power ().

**Код программы**#include <iostream>

using namespace std;

double power(double n, int p = 2) {

double result = n;

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

result \*= n;

}

return result;

}

char power(char n, int p = 2) {

char result = n;

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

result \*= n;

}

return result;

}

short int power(short int n, int p = 2) {

short int result = n;

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

result \*= n;

}

return result;

}

long int power(long int n, int p = 2) {

long int result = n;

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

result \*= n;

}

return result;

}

float power(float n, int p = 2) {

float result = n;

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

result \*= n;

}

return result;

}

int main()

{

setlocale(LC\_ALL, "Russian");

double inputDouble, result;

int powerDigit, answer;

char inputChar, resultChar;

short int inputShortInt, resultShortInt;

long int inputLongInt, resultLongInt;

float inputFloat, resultFloat;

cout << "Введите:\n" << "double: ";

cin >> inputDouble;

cout << "char: ";

cin >> inputChar;

cout << "short int: ";

cin >> inputShortInt;

cout << "long int: ";

cin >> inputLongInt;

cout << "float: ";

cin >> inputFloat;

cout << "Хотите ли ввести степень? (>=1 - да, 0 - нет): ";

cin >> answer;

if (answer) {

while (true) {

cout << "Введите степень: ";

cin >> powerDigit;

if (powerDigit > 2) break;

else cout << "Степень должна быть больше 2!\n";

}

result = power(inputDouble, powerDigit);

resultChar = power(inputChar, powerDigit);

resultShortInt = power(inputShortInt, powerDigit);

resultLongInt = power(inputLongInt, powerDigit);

resultFloat = power(inputFloat, powerDigit);

}

else {

result = power(inputDouble);

resultChar = power(inputChar);

resultShortInt = power(inputShortInt);;

resultLongInt = power(inputLongInt);

resultFloat = power(inputFloat);

}

cout << "Результат:\n" << "double: " << result << endl;

cout << "char: " << resultChar << endl;

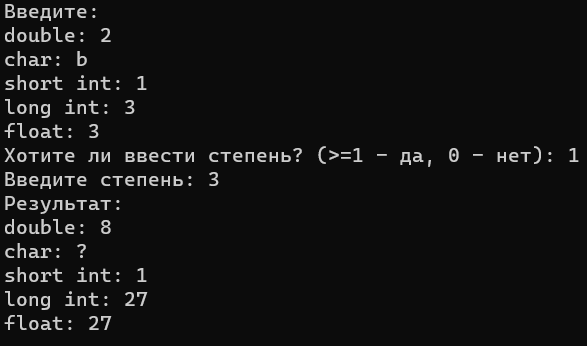
cout << "short int: " << resultShortInt << endl;

cout << "long int: " << resultLongInt << endl;

cout << "float: " << resultFloat << endl;

}

**Проверка выполнения**

****

**Блох схема**



**Задание 3**

Информация по работникам N заводов задается структурой с полями: фамилия, возраст, специальность, средний оклад. Ввести информацию по заводам, посчитать количество слесарей и токарей. Вывести эти значения на консоль.

**Код программы**

#include <iostream>

#include <string>

#include <vector>

#include <windows.h>

using namespace std;

class Worker {

private:

string surname;

int age;

string specialty;

double averageSalary;

public:

Worker(const string& s, int a, const string& sp, double sal) : surname(s), age(a), specialty(sp), averageSalary(sal) {}

string getSurname() const { return surname; }

int getAge() const { return age; }

string getSpecialty() const { return specialty; }

double getAverageSalary() const { return averageSalary; }

};

class FactoryManager {

private:

vector<vector<Worker>> factories;

public:

void addWorkers(int factoryIndex) {

int numWorkers;

cout << "Введите количество работников на заводе " << factoryIndex + 1 << ": ";

cin >> numWorkers;

vector<Worker> factoryWorkers;

for (int j = 0; j < numWorkers; ++j) {

string surname, specialty;

int age;

double averageSalary;

cout << "Введите фамилию работника " << j + 1 << ": ";

cin >> surname;

cout << "Введите возраст работника " << j + 1 << ": ";

cin >> age;

cout << "Введите специальность работника " << j + 1 << " (слесарь/токарь): ";

cin >> specialty;

cout << "Введите средний оклад работника " << j + 1 << ": ";

cin >> averageSalary;

factoryWorkers.emplace\_back(surname, age, specialty, averageSalary);

}

factories.push\_back(factoryWorkers);

}

void countWorkers(int& numFitters, int& numTurners) {

for (const auto& factory : factories) {

for (const auto& worker : factory) {

if (worker.getSpecialty() == "слесарь") {

numFitters++;

}

else if (worker.getSpecialty() == "токарь") {

numTurners++;

}

}

}

}

};

int main() {

setlocale(LC\_ALL, "Russian");

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

int numFactories;

cout << "Введите количество заводов: ";

cin >> numFactories;

FactoryManager factoryManager;

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

factoryManager.addWorkers(i);

}

int numFitters = 0;

int numTurners = 0;

factoryManager.countWorkers(numFitters, numTurners);

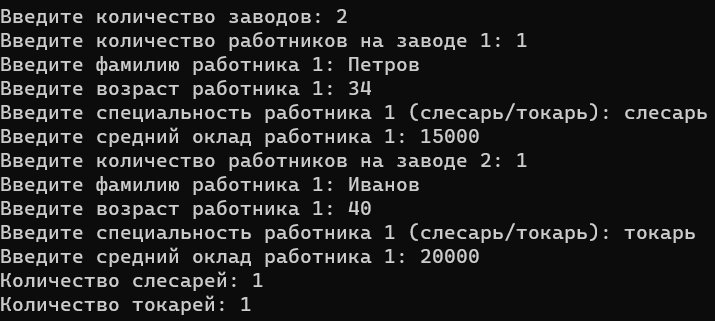
cout << "Количество слесарей: " << numFitters << endl;

cout << "Количество токарей: " << numTurners << endl;

return 0;

}

**Проверка выполнения**

****

**Блок схема**

