Министерство образования и науки Кыргызской Республики

Кыргызский государственный технический университет

им. И.Раззакова

Институт информационных технологий

Кафедра «Программное обеспечение компьютерных систем»

Направление: 710400 «Программная инженерия»

Дисциплина: «Объектно-ориентированное программирование с++»

**ОТЧЕТ**

По лабораторной работе № 1

Тема: «Повторение»

Выполнила: студент группы

ПИ(б)-4-21 Кадырбек кызы Айзат

Проверила: М.Э.Бакытжанович

Бишкек – 2023

**Тема:** Повторить пройденный материал - типы данных, функции и структуры в программировании для лучшего понимания свойств классов в объектно-ориентированном программировании.

**Задания**

***Задача №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 ().

#include <iostream>

using namespace std;

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

int main() {

setlocale(LC\_ALL, "rus");

double a;

int b;

cout << "Введите число: ";

cin >> a;

cout << "Будете вводить степень (1 - да, 0 - нет)? ";

int x;

cin >> x;

if (x) {

cout << "Введите степень: ";

cin >> b;

}

else {

b = 2;

}

cout << "Результат: " << power(a, b) << endl;

return 0;

}

double power(double n, int p) {

double result = 1.0;

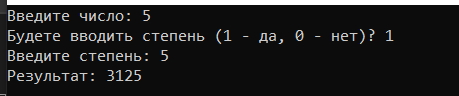
for (int i = 0; i < p; ++i) {

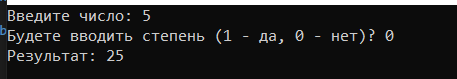
result \*= n;

}

return result;

}





***Задача №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 ().

#include<iostream>

using namespace std;

// Объявление перегруженных функций

double powered(double z, int p);

char powered(char z, int p);

short int powered(short int z, int p);

long int powered(long int z, int p);

float powered(float z, int p);

int main() {

setlocale(LC\_ALL, "rus");

// Выбор типа данных

cout << "Выберите тип данных (1 - double, 2 - char, 3 - short int, 4 - long int, 5 - float): ";

int Type;

cin >> Type;

// Ввод числа

cout << "Введите число: ";

double z;

cin >> z;

// Выбор ввода степени

cout << "Будете вводить степень (1 - да, 0 - нет)? ";

int x;

cin >> x;

int p; // Степень

if (x) {

cout << "Введите степень: ";

cin >> p;

}

else {

p = 2;

}

// Вывод результата

switch (Type) {

case 1:

cout << "Результат: " << powered(z, p) << endl;

break;

case 2:

cout << "Результат: " << static\_cast<int>(powered(static\_cast<char>(z), p)) << endl;

break;

case 3:

cout << "Результат: " << powered(static\_cast<short int>(z), p) << endl;

break;

case 4:

cout << "Результат: " << powered(static\_cast<long int>(z), p) << endl;

break;

case 5:

cout << "Результат: " << powered(static\_cast<float>(z), p) << endl;

break;

default:

cout << "Выбран неверный тип данных." << endl;

}

system("pause");

return 0;

}

// Определение перегруженных функций

double powered(double z, int p) {

double result = 1.0;

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

result \*= z;

}

return result;

}

char powered(char z, int p) {

char result = z;

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

result \*= z;

}

return result;

}

short int powered(short int z, int p) {

short int result = z;

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

result \*= z;

}

return result;

}

long int powered(long int z, int p) {

long int result = z;

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

result \*= z;

}

return result;

}

float powered(float z, int p) {

float result = z;

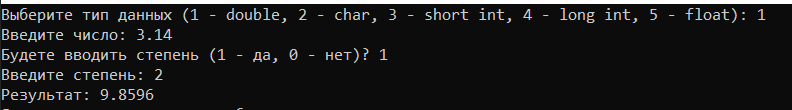
for (int i = 1; i < p; ++i) {

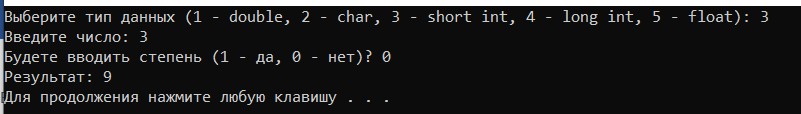
result \*= z;

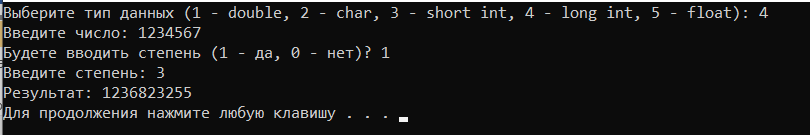
}

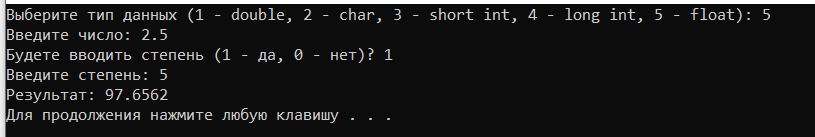
return result;

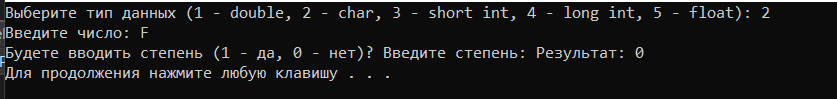
}











***Задача №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.

#include <iostream>

#include <vector>

using namespace std;

struct Worker {

string fam;

int age;

string spec;

int sro;

};

int main() {

setlocale(0, "");

int numFactories;

cout << "Введите количество заводов: ";

cin >> numFactories;

int numTokars = 0, numSlesars = 0;

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

cout << "Введите информацию по " << i + 1 << "-ому заводу:" << endl;

int numWorkers;

cout << "Введите количество работников: ";

cin >> numWorkers;

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

Worker worker;

cout << "Фамилия: "; cin >> worker.fam;

cout << "Возраст: "; cin >> worker.age;

cout << "Специальность (slesar/tokar): "; cin >> worker.spec;

cout << "Средний оклад: "; cin >> worker.sro;

if (worker.spec == "slesar") {

numSlesars++;

}

else if (worker.spec == "tokar") {

numTokars++;

}

}

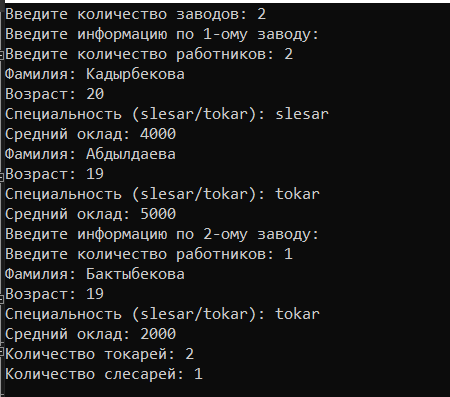
}

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

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

return 0;

}

****