**Федеральное государственное автономное образовательное учреждение высшего образования**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**(ВЫСШАЯ ШКОЛА ПЕЧАТИ И МЕДИАИНДУСТРИИ)**

**(Факультет информационных технологий)**

***(Институт Принтмедиа и информационных технологий)***

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА № 3**

**Дисциплина:** Введение в программирование

**Тема:** Структуры и перечисления

**Выполнил: студент группы 231-338**

Шаура Илья Максимович

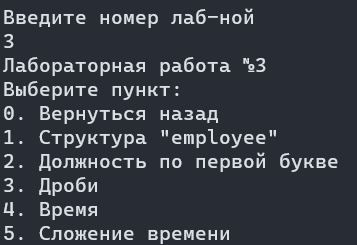
**Дата, подпись** 16.10.2023



**Москва2023**

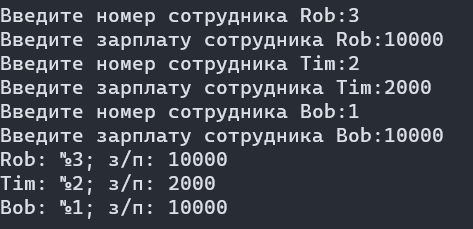
Меню выбора лабораторной / пункта





1. Создайте структуру с именем employee, содержащую два поля: номер сотрудника типа int и величину его зарплаты в рублях типа float. Запросите с клавиатуры данные о трех сотрудниках, сохраните их в трех структурных переменных типа employee и выведите информацию о каждом из сотрудников на экран

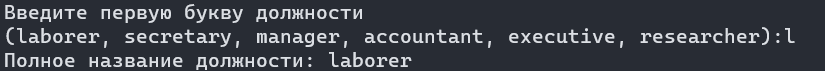




void lab3\_1() {  
 struct employee {  
 int employee\_id;  
 float salary;  
 };  
  
 employee Rob{0, 0}, Tim{0, 0}, Bob{0, 0};  
  
 std::cout << "Введите номер сотрудника Rob:";  
 std::cin >> Rob.employee\_id;  
 std::cout << "Введите зарплату сотрудника Rob:";  
 std::cin >> Rob.salary;  
 std::cout << "Введите номер сотрудника Tim:";  
 std::cin >> Tim.employee\_id;  
 std::cout << "Введите зарплату сотрудника Tim:";  
 std::cin >> Tim.salary;  
 std::cout << "Введите номер сотрудника Bob:";  
 std::cin >> Bob.employee\_id;  
 std::cout << "Введите зарплату сотрудника Bob:";  
 std::cin >> Bob.salary;  
  
 std::cout << "Rob: №" << Rob.employee\_id << "; з/п: " << Rob.salary << std::endl;  
 std::cout << "Tim: №" << Tim.employee\_id << "; з/п: " << Tim.salary << std::endl;  
 std::cout << "Bob: №" << Bob.employee\_id << "; з/п: " << Bob.salary << std::endl;  
 std::cout << std::endl;  
}

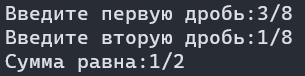
2. Напишите программу, которая сначала по первой букве должности, введенной пользователем, определяет соответствующее значение переменной, помещает это значение в переменную типа etype, а затем выводит полностью название должности, первую букву которой ввел пользователь.





void lab3\_2() {  
 int num;  
 char first\_letter;  
 enum e\_type { *laborer*, *secretary*, *manager*, *accountant*, *executive*, *researcher* };  
  
 std::cout << "Введите первую букву должности\n"  
 "(laborer, secretary, manager, accountant, executive, researcher):";  
 std::cin >> first\_letter;  
  
 switch (first\_letter) {  
 case 'l':  
 num = *laborer*;  
 break;  
 case 's':  
 num = *secretary*;  
 break;  
 case 'm':  
 num = *manager*;  
 break;  
 case 'a':  
 num = *accountant*;  
 break;  
 case 'e':  
 num = *executive*;  
 break;  
 case 'r':  
 num = *researcher*;  
 break;  
 default:  
 num = -1;  
 break;  
 }  
  
 std::cout << "Полное название должности: ";  
 switch (num) {  
 case *laborer*:  
 std::cout << "laborer" << std::endl;  
 break;  
 case *secretary*:  
 std::cout << "secretary" << std::endl;  
 break;  
 case *manager*:  
 std::cout << "manager" << std::endl;  
 break;  
 case *accountant*:  
 std::cout << "accountant" << std::endl;  
 break;  
 case *executive*:  
 std::cout << "executive" << std::endl;  
 break;  
 case *researcher*:  
 std::cout << "researcher" << std::endl;  
 break;  
 default:  
 std::cout << "отсутствует в списке" << std::endl;  
 break;  
 }  
}

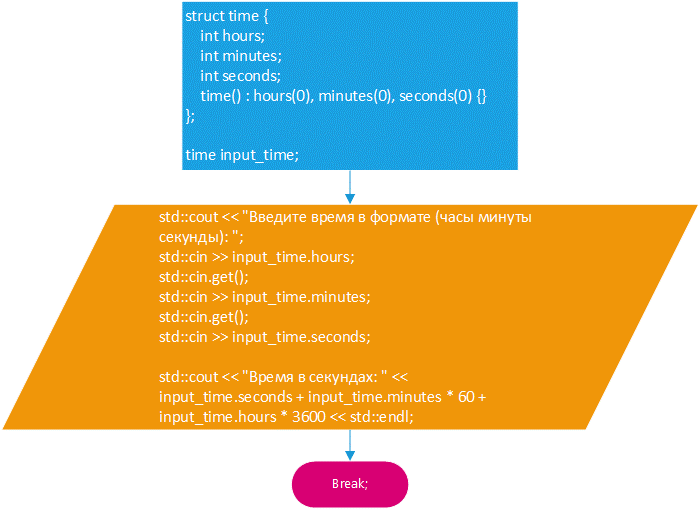
3. Вернитесь к упражнению 3 ЛР1. В этом упражнении требуется написать программу, которая хранит значения двух дробей в виде числителя и знаменателя, а затем складывает эти дроби согласно арифметическому правилу. Измените эту программу так, чтобы значения дробей хранились в структуре fraction, состоящей из двух полей типа int. предназначенных для хранения числителя и знаменателя. Все значения дробей должны храниться в переменных типа fraction.



void lab3\_3() {  
 struct fraction {  
 [[nodiscard]] std::string full\_fraction() const {  
 return std::to\_string(numerator) + '/' + std::to\_string(denominator);  
 }  
 int numerator;  
 int denominator;  
 std::string full\_fraction\_s;  
  
 fraction() {  
 numerator = 0;  
 denominator = 1;  
 }  
 };  
 fraction first, second, result;  
 std::vector<std::string> first\_s, second\_s;  
 std::cout << "Введите первую дробь:";  
 std::cin >> first.full\_fraction\_s;  
 std::cout << "Введите вторую дробь:";  
 std::cin >> second.full\_fraction\_s;  
 first\_s = customSplit(first.full\_fraction\_s, '/');  
 second\_s = customSplit(second.full\_fraction\_s, '/');  
 first.numerator = std::stoi(first\_s[0]);  
 first.denominator = std::stoi(first\_s[0]);  
 first.numerator = std::stoi(first\_s[0]);  
 std::reverse(first\_s.begin(), first\_s.end());  
 first.denominator = std::stoi(first\_s[0]);  
 second.numerator = std::stoi(second\_s[0]);  
 std::reverse(second\_s.begin(), second\_s.end());  
 second.denominator = std::stoi(second\_s[0]);  
 int left, right, nod = 1;  
 left = first.numerator \* second.denominator + first.denominator \* second.numerator;  
 right = first.denominator \* second.denominator;  
  
 for (int i = left; i > 0; i--) {  
 if (left % i == 0 && right % i == 0) {  
 nod = i;  
 break;  
 }  
 }  
 left /= nod;  
 right /= nod;  
 result.numerator = left;  
 result.denominator = right;  
  
 std::string result\_s = result.full\_fraction();  
 if (left == right) result\_s = std::to\_string(left);  
 std::cout << "Сумма равна:" << result\_s << std::endl;  
}



4. Создайте структуру с именем time. Три ее поля, имеющие тип int. будут называться hours, minutes и seconds. Напишите программу, которая просит пользователя ввести время в формате часы, минуты, секунды. Можно запрашивать на ввод как три значения сразу, так и выводить для каждой величины отдельное приглашение.

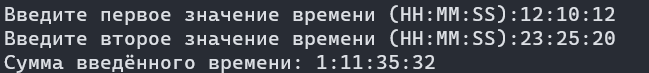




void lab3\_4() {  
 struct time {  
 int hours;  
 int minutes;  
 int seconds;  
 time() : hours(0), minutes(0), seconds(0) {}  
 };  
  
 time input\_time;  
  
 std::cout << "Введите время в формате (часы минуты секунды): ";  
 std::cin >> input\_time.hours;  
 std::cin.get();  
 std::cin >> input\_time.minutes;  
 std::cin.get();  
 std::cin >> input\_time.seconds;  
  
 std::cout << "Время в секундах: " << input\_time.seconds + input\_time.minutes \* 60 + input\_time.hours \* 3600 << std::endl;  
}

5. Используя структуру time из предыдущего упражнения, напишите программу, которая получает от пользователя два значения времени в формате 12:59:59, сохраняет их в переменных типа struct time, затем переводит оба значения в секунды, складывает их, переводит сумму в исходный формат, сохраняет его в переменной типа time и выводит полученный результат на экран в формате 12:59:59.





void lab3\_5() {  
 struct time {  
 int days;  
 int hours;  
 int minutes;  
 int seconds;  
 time() : days(0), hours(0), minutes(0), seconds(0) {}  
 };  
  
 time first\_time, second\_time, sum\_time;  
  
 std::cout << "Введите первое значение времени (HH:MM:SS):";  
  
 std::cin >> first\_time.hours;  
 std::cin.get();  
 std::cin >> first\_time.minutes;  
 std::cin.get();  
 std::cin >> first\_time.seconds;  
  
 std::cout << "Введите второе значение времени (HH:MM:SS):";  
  
 std::cin >> second\_time.hours;  
 std::cin.get();  
 std::cin >> second\_time.minutes;  
 std::cin.get();  
 std::cin >> second\_time.seconds;  
  
 sum\_time.hours = first\_time.hours + second\_time.hours;  
 sum\_time.minutes = first\_time.minutes + second\_time.minutes;  
 sum\_time.seconds = first\_time.seconds + second\_time.seconds;  
  
 while (sum\_time.seconds > 59) {  
 sum\_time.minutes++;  
 sum\_time.seconds -= 60;  
 }  
 while (sum\_time.minutes > 59) {  
 sum\_time.hours++;  
 sum\_time.minutes -= 60;  
 }  
 while (sum\_time.hours > 23) {  
 sum\_time.days++;  
 sum\_time.hours -= 24;  
 }  
  
 std::cout << "Сумма введённого времени: " << sum\_time.days << ':' << sum\_time.hours << ':' << sum\_time.minutes << ':' << sum\_time.seconds << std::endl;  
  
}