Министерство образования Республики Беларусь

Учреждение образования

“Брестский государственный технический университет”

Кафедра ИИТ

**Лабораторная работа №10**

По дисциплине: “Основы алгоритмизации и программирования”

Тема: ”Структуры, перечисления, объединения”

Вариант №6

**Выполнил:**

Студент 1-го курса

Группы ПО-7

Комиссаров А.Е.

**Проверила:**

Войцехович О.Ю.

Брест, 2020

**Цель работы:** Изучить синтаксис и правила работы со структурами. Реализовать программу с применением структур, перечислений и объединений.

**Задание:** Информация о сотрудниках фирмы включает ФИО, табельный номер, количество отработанных часов за месяц, почасовой тариф. Вывести размер заработной платы каждого сотрудника.

**Код программы:**

#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

struct worker

{

string fio;

union{int tabnum;}u\_tabnum;

int tarif;

float time;

};

enum tarif\_type { low = 1, avg = 2, high = 3 };

int gettarif(int i, worker\* work) {

int val = work[i].tarif;

if (val == 1 || val == 2 || val == 3) {

if (val == low) { return 5; }

if (val == avg) { return 25; }

if (val == high) { return 50; }

}

else return val;

}

int main()

{

setlocale(LC\_ALL, "Russian");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int choice, n = 0, del;

worker\* work = new worker[100];

do

{

system("cls");

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

cout << "Рабочий №" << i + 1 << " - ";

cout << "ФИО:" << work[i].fio << ", тариф:" << gettarif(i, work) << ", время работы: " << work[i].time << ", табельный №: " << work[i].u\_tabnum.tabnum << ";" << endl;

}

cout << endl << "Меню: " << endl;

cout << "1-Ввод данных;" << endl;

cout << "2-Вывод данных;" << endl;

cout << "3-Сортировка по кол-ву часов;" << endl;

cout << "4-Удаление по заданному значению поля;" << endl;

cout << "5-Зароботная плата сотрудников;" << endl;

cout << "6-Выход." << endl;

cout << endl << "Ваш выбор: ";

cin >> choice;

switch (choice)

{

case 1: {

system("cls");

cout << "Введите количество рабочих: ";

cin >> n;

worker\* work2 = new worker[n];

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

cout << "[" << i + 1 << "] - - - - - - - - - - - - - - -" << endl;

cout << "Введите ФИО рабочего: "; cin.ignore(); getline(cin, work2[i].fio);

cout << "Введите табельный номер рабочего: "; cin >> work2[i].u\_tabnum.tabnum;

cout << "Введите тариф (1 - 5руб/ч, 2 - 25руб/ч, 3 - 50руб/ч): "; cin >> work2[i].tarif;

cout << "Введите время работы: "; cin >> work2[i].time;

work[i] = work2[i];

}

delete[] work2;

} break;

case 2: {

system("cls");

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

cout << "Рабочий №" << i + 1 << " - ";

cout << "ФИО:" << work[i].fio << ", тариф:" << gettarif(i, work) << ", время работы: " << work[i].time << ", табельный №: " << work[i].u\_tabnum.tabnum << ";" << endl;

}

} break;

case 3: {

system("cls");

for (int i = n - 1; i >= 0; i--)

{

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

{

if (work[j].time > work[j + 1].time)

{

worker temp = work[j];

\*(work + j) = \*(work + j + 1);

\*(work + j + 1) = temp;

}

}

}

} break;

case 4: {

system("cls");

cout << "Введите кол-во часов при котором стоит удалить сотрудника(-ов):"; cin >> del;

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

if (work[i].time == del) { for (int j = i; j < n; j++) { work[j] = work[j + 1]; } i--; n--; }

}

}break;

case 5:{

system("cls");

cout << "Зарплата сотрудников фирмы:" << endl;

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

cout << "Рабочий №" << i + 1 << " - ";

cout << "ФИО:" << work[i].fio << ", зарплата:" << gettarif(i, work) \* work[i].time << ";" << endl;

}

}break;

case 6:{

cout << "Выход из программы..." << endl;

}break;

default:{cout << "Неверный ввод!" << endl;}

}

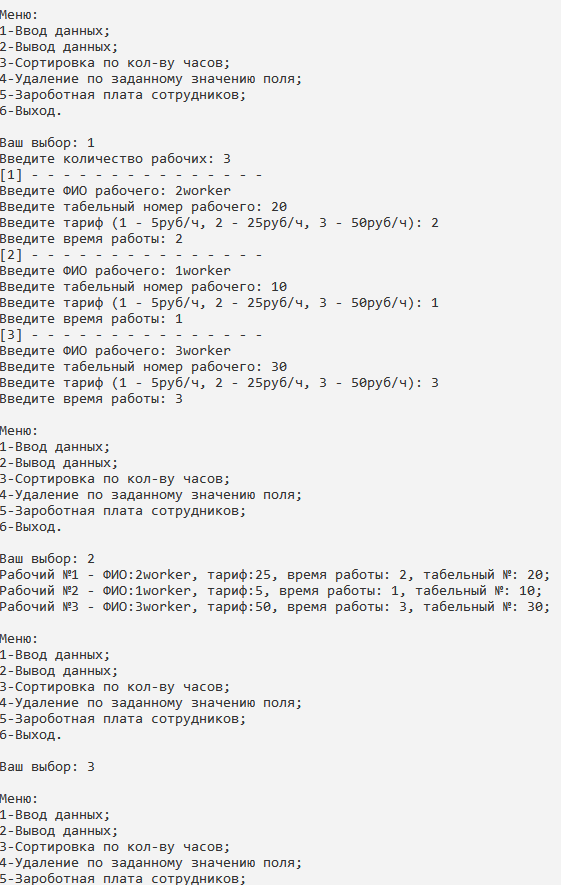
} while (choice != 6);

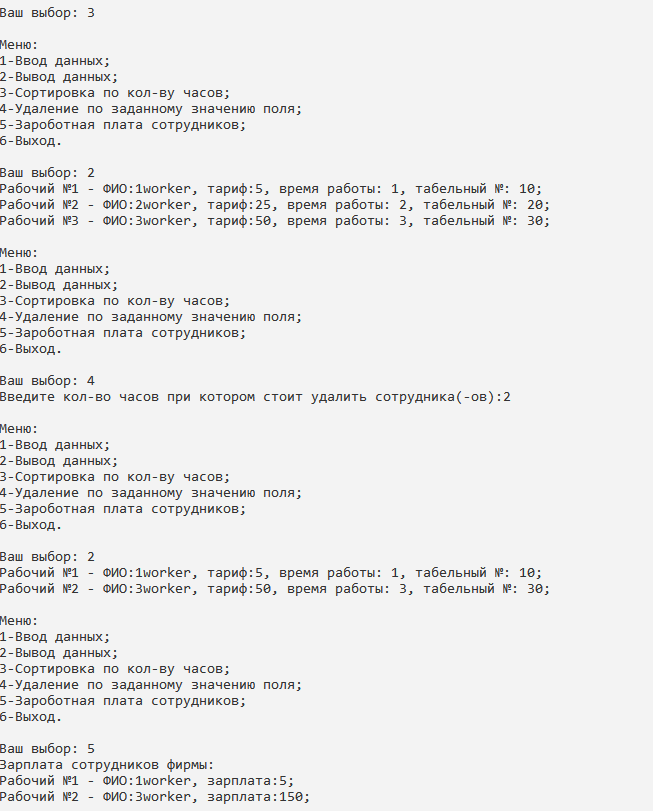
delete[]work;

return 0;

}

**Результат выполнения:**

****

****

****

**Вывод:** изучил синтаксис и правила работы со структурами. Реализовал программу с применением структур, перечислений и объединений.