Лабораторная работа №9

**Тема:** Разработка программ с использованием структур и перечислений.

**Цель:** Научиться разрабатывать алгоритмы и реализовать программы с применением переменных структур и перечислений.

**Техническое оснащение:** MS Windows 10, MS Word 2019, MS VS 2019.

Выполнение работы

Вариант 3

**Пример 1.** Программа с использованием перечисления (рисунок 1).

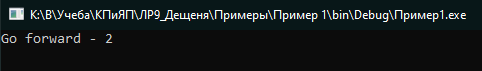


Рисунок 1 – Выполнение примера 1

**Пример 2.** Программа с использованием перечисления цветов (рисунок 2).

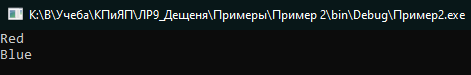


Рисунок 2 – Выполнение примера 2

**Пример 3.** Создание структуры без оператора new (рисунок 3).



Рисунок 3 – Выполнение примера 3

**Пример 4.** Создание структуры с оператором new, без параметров (рисунок 4).

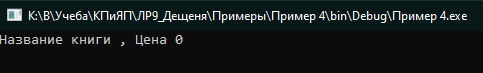


Рисунок 4 – Выполнение примера 4

**Пример 5.** Создание структуры с оператором new, с параметрами (рисунок 5).



Рисунок 5 – Выполнение примера 5

**Задание.** Перечень основных дисциплин:

Название дисциплины; кафедра, на которой читается дисциплина; ФИО преподавателя, читающего лекции; ФИО преподавателя, ведущего лабораторные

занятия.

**Запрос**. Вывести на экран дисциплины, которые читаются преподавателями заданной кафедры.

Форма программы (рисунок 6).

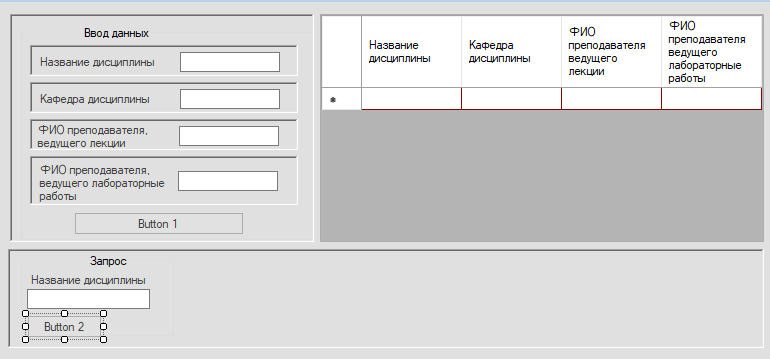


Рисунок 6 – Исходная форма программы

Изменяем названия кнопок, цвет DataGrid и название самой формы (рисунок 7).

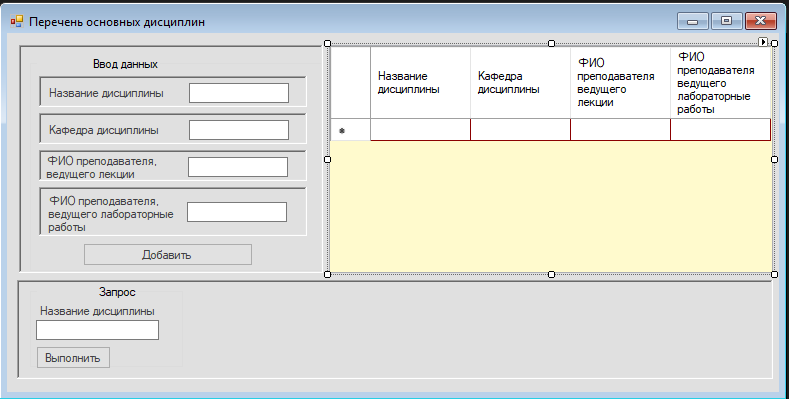


Рисунок 7 – Преобразованная форма программы

Листинг программы

Тестирование программы. Добавления записи (рисунок 8).

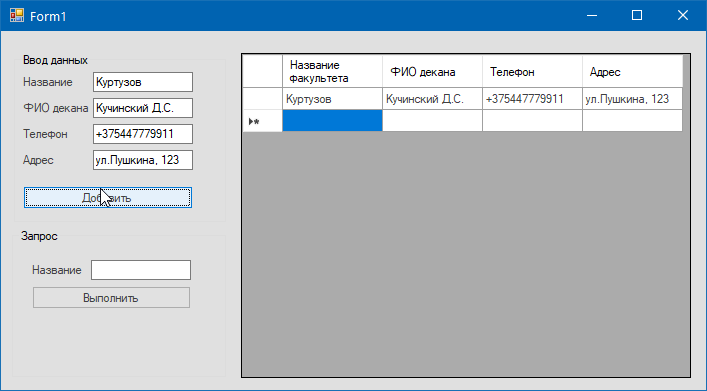


Рисунок 8 – Добавление новой записи

Листинг программы

|  |
| --- |
| using System;  using System.Windows.Forms;  namespace Задание  {  public partial class Form1 : Form  {  public Form1()  {  InitializeComponent();  }  struct facult {  public string name;  public string decan;  public string telefon;  public string adress;  public facult(string name, string decan, string telefon, string adress)  {  this.name = name;  this.decan = decan;  this.telefon = telefon;  this.adress = adress;  }  public string[] ToParam()  {  string[] arr = { name, decan, telefon, adress};  return arr;  }  }  private void button1\_Click(object sender, EventArgs e)  {  facult NewFacult = new facult(textBox1.Text, textBox6.Text, textBox7.Text, textBox8.Text);  dataGridView1.Rows.Add(NewFacult.ToParam());  }  private void button2\_Click(object sender, EventArgs e)  {  string name = textBox5.Text;  foreach (DataGridViewRow i in dataGridView1.Rows)  {  string rowname = i.Cells[0].Value?.ToString() ?? "";  if (rowname == name)  {  label7.Text = $"Кафедра Дисциплины: {i.Cells[1].Value}.";  label8.Text = $"ФИО преподавателя ведущего лекции: {i.Cells[2].Value}";  label9.Text = $"ФИО преподавателя ведущего лабораторные работы: {i.Cells[3].Value}";  return;  }  }  label7.Text = $"Не удалось найти дисциплину под названием: {name}";  }    }  } |

Запускаем программу, вводим данные о дисциплинах и проверяем запрос (рисунок 9).

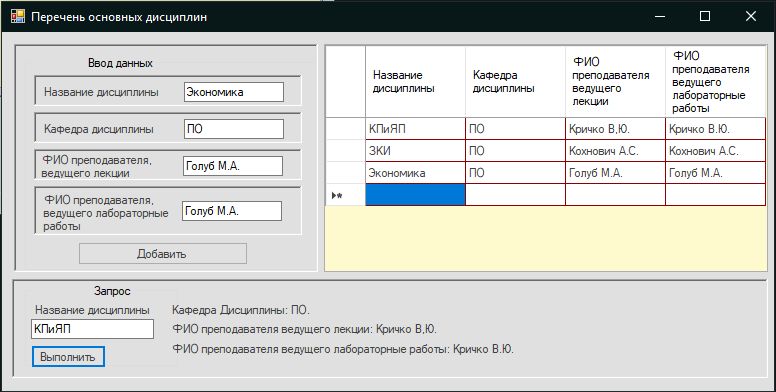


Рисунок 9 – Выполнение корректного запроса

**Вывод:** при выполнении лабораторной работы получил практические навыки по использованию структур и перечислений.