



ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ
ГОРОДА МОСКВЫ
Государственное бюджетное профессиональное
образовательное учреждение города Москвы
«Колледж малого бизнеса № 4»
(ГБПОУ КМБ № 4)

Лабораторная работа Windows Form №5

Специальность: 09.02.07 Информационные системы и программирование

Форма обучения: очная

Студентка: Чучелина Таисия Сергеевна

Группа: ИПО-22.24

Руководитель: Рыбаков Александр Сергеевич

Отчётная работа защищена с оценкой «___» _____

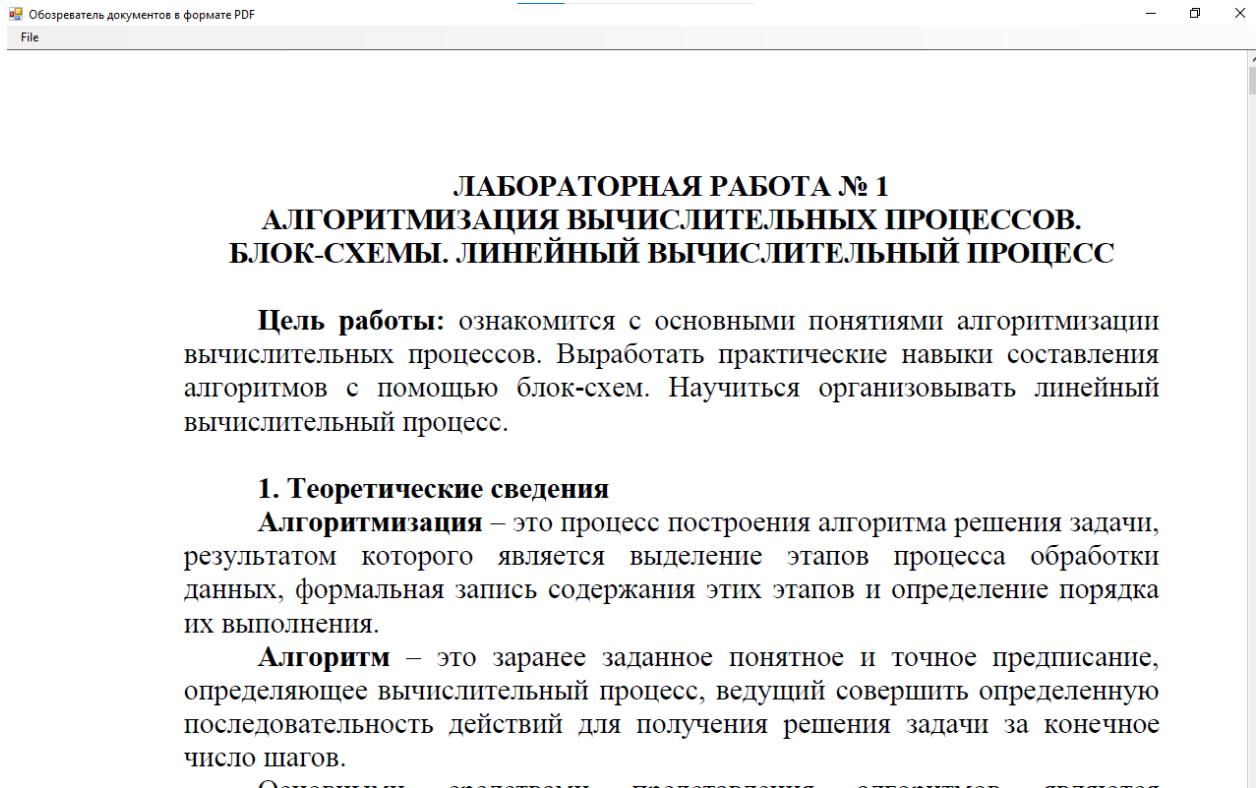
Москва, 2025 г.

Оглавление

Лабораторная работа №5	3
Упражнение 1	3
Упражнение 2	4

Лабораторная работа №5

Упражнение 1



Цель работы: ознакомится с основными понятиями алгоритмизации вычислительных процессов. Выработать практические навыки составления алгоритмов с помощью блок-схем. Научиться организовывать линейный вычислительный процесс.

1. Теоретические сведения

Алгоритмизация – это процесс построения алгоритма решения задачи, результатом которого является выделение этапов процесса обработки данных, формальная запись содержания этих этапов и определение порядка их выполнения.

Алгоритм – это заранее заданное понятное и точное предписание, определяющее вычислительный процесс, ведущий совершение определенную последовательность действий для получения решения задачи за конечное число шагов.

Основные определения: алгоритмизация, алгоритм, блок-схема

5.1 Работа формы по открытию файлов PDF формата

Код:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

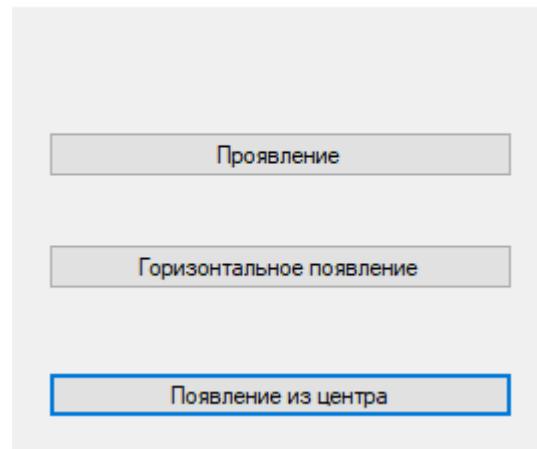
namespace PDF_Reader
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void openToolStripMenuItem_Click(object sender,
System.EventArgs e)
        {
            openFileDialog1.Filter = "Файлы pdf|*.pdf";
            openFileDialog1.ShowDialog();
            axAcroPDF1.LoadFile(openFileDialog1.FileName);
        }
    }
}
```

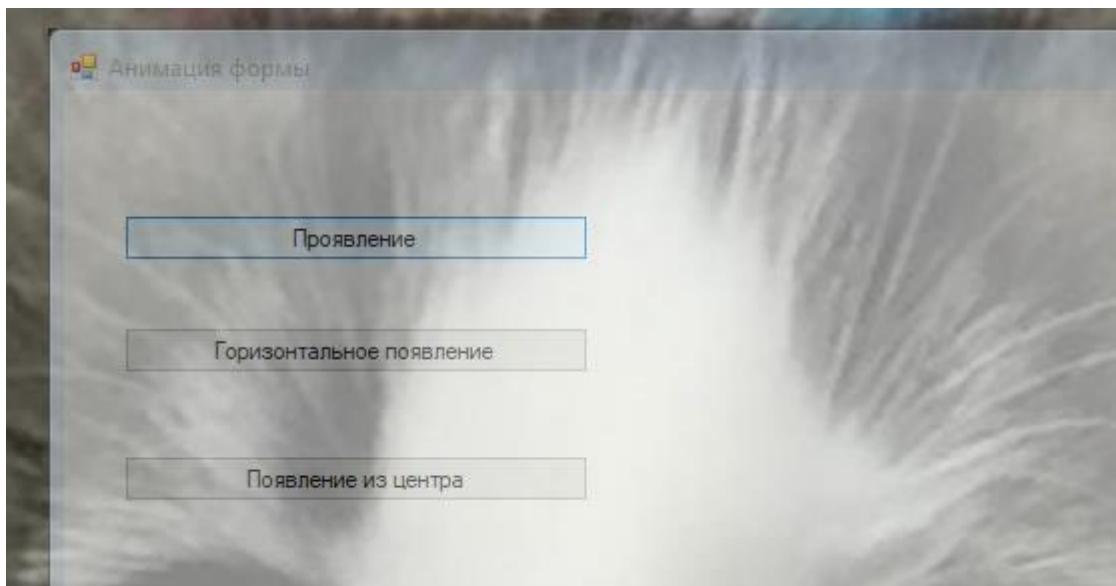
```
private void exitToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.Close();
}
```

Упражнение 2

Анимация формы

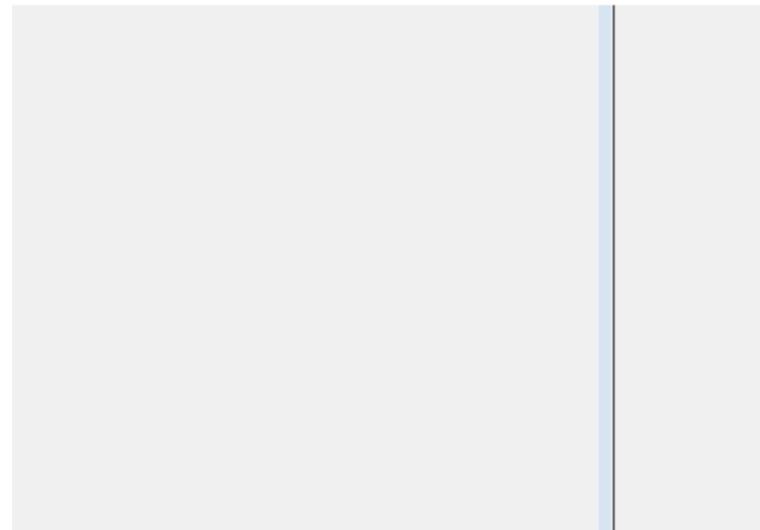


2.1 Работа формы анимации



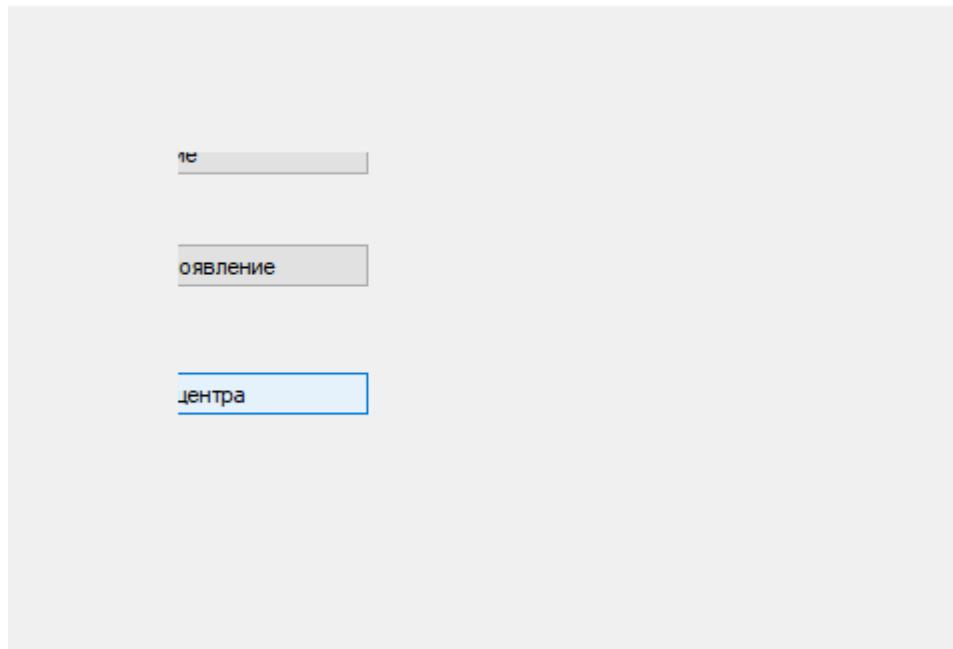
2.2 Анимация проявления

формы



2.3 Анимация горизонтального появления

я формы



2.4 Анимация появления из центра

Код:

Form1.cs:

```
using AnimatedWindow;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WinAnim
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {

        }

        private void btnAW_BLEND_Click_1(object sender, EventArgs e)
        {
            // Скрываем окно
            this.Hide();
            // Запускаем анимацию.
            // Второй параметр в скобках — время анимации в
            // миллисекундах.
            WinAPIClass.AnimateWindow(this, 3000,
WinAPIClass.AnimateWindowFlags.AW_ACTIVATE |
WinAPIClass.AnimateWindowFlags.AW_BLEND);
            // Отображаем кнопки после анимации
            this.btnAW_BLEND.Invalidate();
            this.btnHOR_AW_SLIDE.Invalidate();
            this.btnCenter_AW_SLIDE.Invalidate();
        }

        private void btnHOR_AW_SLIDE_Click_1(object sender, EventArgs e)
        {
            this.Hide();
            WinAPIClass.AnimateWindow(this, 3000,
WinAPIClass.AnimateWindowFlags.AW_HOR_POSITIVE |
WinAPIClass.AnimateWindowFlags.AW_SLIDE);
            this.btnAW_BLEND.Invalidate();
            this.btnHOR_AW_SLIDE.Invalidate();
            this.btnCenter_AW_SLIDE.Invalidate();
        }

        private void btnCenter_AW_SLIDE_Click_1(object sender, EventArgs e)
        {
            this.Hide();
            WinAPIClass.AnimateWindow(this, 3000,
WinAPIClass.AnimateWindowFlags.AW_CENTER |
WinAPIClass.AnimateWindowFlags.AW_SLIDE);
```

```

        this.btnClose_AW_BLEND.Invalidate();
        this.btnCloseHOR_AW_SLIDE.Invalidate();
        this.btnCloseCenter_AW_SLIDE.Invalidate();
    }
}

```

Создание класса:

WinApiClass.cs:

```

using System;
using System.Runtime.InteropServices;
using System.Windows.Forms;
namespace AnimatedWindow
{
    public class WinAPIClass
    {
        #region Анимация окна
        /// <summary>
        /// Тип анимации окна. Перечисление возвращает тип int
        /// после приведения. Каждому элементу присвоено
        /// свое значение типа int.
        /// </summary>
        [Flags]
        public enum AnimateWindowFlags : int
        {
            AW_HOR_POSITIVE = 1,
            AW_HOR_NEGATIVE = 2,
            AW_VER_POSITIVE = 4,
            AW_VER_NEGATIVE = 8,
            AW_CENTER = 16,
            AW_HIDE = 65536,
            AW_ACTIVATE = 131072,
            AW_SLIDE = 262144,
            AW_BLEND = 524288
        };
        /// <summary>
        /// Анимация окна.
        /// </summary>
        /// <param name="hwnd">Окно.</param>
        /// <param name="dwTime">Время.</param>
        /// <param name="dwFlags">Тип анимации. Если в
        /// неуправляемом коде используется перечисление, то его
        /// нужно конвертировать в тип данных int.</param>
        /// <returns></returns>
        [DllImportAttribute("user32.dll", EntryPoint = "AnimateWindow", SetLastError = true)]
        public static extern bool AnimateWindow(IntPtr hwnd, int dwTime,
        int dwFlags);
        /// <summary>
        /// Анимация окна.
        /// </summary>
        /// <param name="ctrl">Окно.</param>
        /// <param name="dwTime">Время.</param>
        /// <param name="Flags">Флаги.</param>
        /// <returns></returns>
        public static bool AnimateWindow(Control ctrl, int dwTime,
        AnimateWindowFlags Flags)
        {
            return AnimateWindow(ctrl.Handle, dwTime, (int)Flags);
        }
        #endregion
    }
}

```

}