



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

**Рубцовский индустриальный институт (филиал)**  
федерального государственного бюджетного образовательного  
учреждения высшего образования  
«Алтайский государственный технический университет им. И.И. Ползунова»  
(РИИ АлтГТУ)

**Е.А. ДУДНИК**

## **ПРОЕКТИРОВАНИЕ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ**

Методические указания для самостоятельной работы студентам  
направления подготовки «Информатика и вычислительная техника»

Рубцовск 2021

## ББК 81.2

Дудник, Е.А. Проектирование пользовательских интерфейсов: методические указания для самостоятельной работы студентам направления подготовки «Информатика и вычислительная техника» /Е.А. Дудник; Рубцовский индустриальный институт. – Рубцовск: РИИ, 2021. – 22 с. [ЭР].

Данные учебно-методические рекомендации предназначены для студентов направлений подготовки «Информатика и вычислительная техника», изучающих дисциплину «Проектирование пользовательских интерфейсов». Представлен лабораторный практикум.

Рассмотрены и одобрены на заседании кафедры ПМ Рубцовского индустриального института.  
Протокол № 9 от 18.03.2021 г.

## 1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Компетенция	Содержание компетенции	Индикатор	Содержание индикатора
ПК-3	Способен проектировать пользовательские интерфейсы по готовому образцу или концепции интерфейса	ПК-3.1	Проектирует интерфейс по концепции или по образцу уже спроектированной части интерфейса

## 2. Место дисциплины в структуре образовательной программы

Дисциплины (практики), предшествующие изучению дисциплины, результаты освоения которых необходимы для освоения данной дисциплины.	Программирование
Дисциплины (практики), для которых результаты освоения данной дисциплины будут необходимы, как входные знания, умения и владения для их изучения.	Выпускная квалификационная работа, Командная разработка программного продукта, Преддипломная практика

## 3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося

3.1. Очная форма обучения: Общий объем дисциплины в з.е. /час: 3 / 108

Форма промежуточной аттестации: Экзамен

Форма обучения	Виды занятий, их трудоемкость (час.)				Объем контактной работы обучающегося с преподавателем (час)
	Лекции	Лабораторные работы	Практические занятия	Самостоятельная работа	
очная	16	16	0	76	38

3.2. Заочная форма обучения: Общий объем дисциплины в з.е. /час: 3 / 108

Форма промежуточной аттестации: Экзамен

Форма обучения	Виды занятий, их трудоемкость (час.)				Объем контактной работы обучающегося с преподавателем (час)
	Лекции	Лабораторные работы	Практические занятия	Самостоятельная работа	
заочная	6	6	0	96	15

**4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий**

**Форма обучения: очная**

**Семестр: 2**

**Лекционные занятия (16ч.)**

- 1. Основы проектирование пользовательского интерфейса(2ч.)[3,5,8]**  
Интерфейс пользователя: мост между человеком и компьютером. Основные принципы разработки пользовательского интерфейса. Стандартизация пользовательского интерфейса.
- 2. Проектирование, ориентированное на цели(2ч.)[3,5,8]** Эволюция проектирования. Планирование и проектирование поведения. Выявление целей пользователей. Целеориентированный процесс проектирования. Выбор структуры диалога. Разработка сценария диалога. Визуальные атрибуты отображаемой информации
- 3. Модели реализации и ментальные модели. Проектирование графического пользовательского интерфейса.(2ч.)[3,5,8]** Модели реализации. Пользовательские ментальные модели. Модели представления. Модели представления механической и информационной эры. Особенности графического интерфейса. Объектный подход к проектированию интерфейса Компоненты графического интерфейса Взаимодействие пользователя с приложением. Общие правила взаимодействия с объектами. Операции пересылки и создания объектов.
- 4. Проектирование для пользователей с различной подготовкой.(2ч.)[3,5,8]** Новички, эксперты и середняки. Качественные и количественные исследования. Как понять пользователей: качественные исследования. Этнографические интервью: интервьюирование и наблюдение за пользователями. Прочие виды исследований
- 5. Проектирование элементов управления(2ч.)[3,5,6,8]** Окна и пиктограммы. Проектирование пиктограмм. Первичные окна. Вторичные окна. Основные свойства вторичных окон. Панели свойств и контроля параметров. Диалоговые панели. Интерпретация системой имени файла. Другие типы вторичных окон. Меню. Кнопки. Переключатели. Флажки. Списки. Текстовые области. Поле назначения горячих клавиш Панель инструментов и строка состояния. Другие элементы графического интерфейса. Выбор визуальных атрибутов отображаемой информации . Композиция и организация. Цвет. Шрифт. Пространственное размещение визуальных элементов. Визуализация выполняемых операций.
- 6. Модели пользователей: персонажи и цели. Проектирование средств поддержки пользователя(2ч.)[3,5,6,8]** Персонажи .Цели. Разработка персонажей. Прочие модели. Окно сообщение. Контекстная помощь. Проблемно-ориентированная помощь. Справочник. Мастера. Средства обучения пользователя. Средства адаптации пользовательского интерфейса.

**7. Основы проектирования: сценарии и требования {лекция с разбором конкретных ситуаций} (2ч.)[3,5,6,8]** Сценарии: повествование как средство проектирования. Требования: информационное обеспечение проектирования взаимодействия. Выработка требований с использованием персонажей и сценариев.

**8. От требований к пользовательскому интерфейсу: общая инфраструктура и детализация {лекция с разбором конкретных ситуаций} (2ч.)[3,5,6,8]** Общая инфраструктура пользовательского интерфейса. Детализация формы и поведения. Проверка результата проектирования и юзабилити' тестирование. Проектирует интерфейс по концепции или по образцу уже спроектированной части интерфейса

#### **Лабораторные работы (16ч.)**

**1. Стандартные элементы пользовательского интерфейса(4ч.)[1,2,4,7]**

**2. Прототипирование и визуализация пользовательского интерфейса(4ч.)[1,2,4,7]**

**3. Диалоговый интерфейс: разработка сценариев и структуры диалога(4ч.)[1,2,4,7]**

**4. Использование графики при создании пользовательского интерфейса. Заставки.(4ч.)[1,2,4,7]**

#### **Самостоятельная работа (76ч.)**

**1. Подготовка к экзамену(36ч.)[1,3,4,5,6]**

**2. Подготовка к выполнению лабораторных работ(16ч.)[1,2,4]**

**3. Изучение теоретического материала(24ч.)[3,4,6,7,8]**

**Форма обучения: заочная**

**Семестр: 3**

#### **Лекционные занятия (6ч.)**

**1. Основы проектирование пользовательского интерфейса(1ч.)[3,5,8]**  
Интерфейс пользователя: мост между человеком и компьютером. Основные принципы разработки пользовательского интерфейса. Стандартизация пользовательского интерфейса.. Эволюция проектирования. Планирование и проектирование поведения. Выявление целей пользователей. Целеориентированный процесс проектирования. Выбор структуры диалога. Разработка сценария диалога. Визуальные атрибуты отображаемой информации.

**2. Модели реализации и ментальные модели. Проектирование графического пользовательского интерфейса.(1ч.)[3,5,8]** Модели реализации. Пользовательские ментальные модели. Модели представления. Модели представления механической и информационной эры. Особенности графического интерфейса. Объектный подход к проектированию интерфейса Компоненты

графического интерфейса Взаимодействие пользователя с приложением. Общие правила взаимодействия с объектами. Операции пересылки и создания объектов.

**3. Проектирование для пользователей с различной подготовкой.(1ч.)[3,5,8]**

Новички, эксперты и середняки. Качественные и количественные исследования. Как понять пользователей: качественные исследования. Этнографические интервью: интервьюирование и наблюдение за пользователями. Прочие виды исследований

**4. Проектирование элементов управления(1ч.)[3,5,6,8]**

Окна и пиктограммы. Проектирование пиктограмм. Первичные окна. Вторичные окна. Основные свойства вторичных окон. Панели свойств и контроля параметров. Диалоговые панели. Интерпретация системой имени файла. Другие типы вторичных окон.

Меню. Кнопки. Переключатели. Флажки. Списки. Текстовые области. Поле назначения горячих клавиш Панель инструментов и строка состояния. Другие элементы графического интерфейса. Выбор визуальных атрибутов отображаемой информации. Композиция и организация. Цвет. Шрифт. Пространственное размещение визуальных элементов. Визуализация выполняемых операций.

**5. Модели пользователей: персонажи и цели. Проектирование средств**

**поддержки пользователя(1ч.)[3,5,6,8]** Персонажи. Цели. Разработка персонажей.

Прочие модели. Окно сообщение. Контекстная помощь. Проблемно-ориентированная помощь. Справочник. Мастера. Средства обучения пользователя. Средства адаптации пользовательского интерфейса.

**6. Основы проектирования: сценарии и требования {лекция с разбором конкретных ситуаций} (1ч.)[3,5,6,8]**

Сценарии: повествование как средство проектирования. Требования: информационное обеспечение проектирования взаимодействия. Выработка требований с использованием персонажей и сценариев. Общая инфраструктура пользовательского интерфейса. Детализация формы и поведения. Проверка результата проектирования и юзабилити' тестирование. Проектирует интерфейс по концепции или по образцу уже спроектированной части интерфейса

**Лабораторные работы (6ч.)**

**1. Стандартные элементы пользовательского интерфейса(1ч.)[1,2,4,7]**

**2. Прототипирование и визуализация пользовательского интерфейса(1ч.)[1,2,4,7]**

**3. Диалоговый интерфейс: разработка сценариев и структуры диалога(2ч.)[1,2,4,7]**

**4. Использование графики при создании пользовательского интерфейса. Заставки.(2ч.)[1,2,4,7]**

**5. Перечень учебно-методического обеспечения самостоятельной работы обучающихся по дисциплине**

Для каждого обучающегося обеспечен индивидуальный неограниченный доступ к электронно-библиотечным системам: Лань, Университетская библиотека он-лайн, электронной библиотеке АлтГТУ и к электронной информационно-образовательной среде:

1. Дудник, Е.А. Программирование. Часть III (лабораторный практикум по программированию в среде Delphi) [текст]: учебно-метод. пособие для студентов направления ИВТ/ Е.А. Дудник, Н.С. Зорина. - Рубцовск: РИИ, 2017. - 63 с. (6 экз.+ЭР)

2. Культин Н.Б. Delphi в задачах и примерах : [текст]. -3-е изд., испр. и доп./ СПб.: БХВ-Петербург, 2012 - 288 с. (10 экз.)

## **6. Перечень учебной литературы**

### **6.1. Основная литература**

3. Мандел, Т. Разработка пользовательского интерфейса / Т. Мандел. — Москва : ДМК Пресс, 2007. — 418 с. — ISBN 5-94074-069-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/1227> (дата обращения: 18.01.2021). — Режим доступа: для авториз. пользователей.

4. Стивенс, Р. Delphi. Готовые алгоритмы : учебное пособие / Р. Стивенс. — Москва : ДМК Пресс, 2007. — 384 с. — ISBN 5-94074-106-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/1234> (дата обращения: 05.02.2021). — Режим доступа: для авториз. пользователей.

### **6.2. Дополнительная литература**

5. Рояк, М.Э. Программирование под Windows графических интерфейсов пользователя : учебное пособие : [16+] / М.Э. Рояк, И.М. Ступаков ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2018. – 72 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=575018> (дата обращения: 15.11.2020). – Библиогр. в кн. – ISBN 978-5-7782-3754-4. – Текст : электронный.

6. Белов, В. В. Программирование в Delphi: процедурное, объектно-ориентированное, визуальное : учебное пособие / В. В. Белов, В. И. Чистякова. — 2-е изд., стер. — Москва : Горячая линия-Телеком, 2017. — 240 с. — ISBN 978-5-9912-0412-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/111017> (дата обращения: 05.02.2021). — Режим доступа: для авториз. пользователей.

## **7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

7.  
[https://www.studmed.ru/mandel-t-razrabotka-polzovatel'skogo-interfeysa\\_1b011418471](https://www.studmed.ru/mandel-t-razrabotka-polzovatel'skogo-interfeysa_1b011418471)

.html

8. [www.books.ru](http://www.books.ru)

Купер А., Рейман Р., Кронин Д. Алан Купер об интерфейсе. Основы проектирования взаимодействия. – Пер. с англ. – СПб.: Символ'Плюс, 2009 – 688 с..

## **8. Фонд оценочных материалов для проведения текущего контроля успеваемости и промежуточной аттестации**

Содержание промежуточной аттестации раскрывается в комплекте контролирующих материалов, предназначенных для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС, которые хранятся на кафедре-разработчике РПД в печатном виде и в ЭИОС.

Фонд оценочных материалов (ФОМ) по дисциплине представлен в приложении А.

## **9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем**

Для успешного освоения дисциплины используются ресурсы электронной информационно-образовательной среды, образовательные интернет-порталы, глобальная компьютерная сеть Интернет. В процессе изучения дисциплины происходит интерактивное взаимодействие обучающегося с преподавателем через личный кабинет студента. Для изучения данной дисциплины профессиональные базы данных и информационно-справочные системы не требуются.

<b>№пп</b>	<b>Используемое программное обеспечение</b>
1	Dev-C++
2	Lazarus
3	LibreOffice
4	Windows
5	Антивирус Kaspersky

<b>№пп</b>	<b>Используемые профессиональные базы данных и информационные справочные системы</b>
1	Бесплатная электронная библиотека онлайн "Единое окно к образовательным ресурсам" для студентов и преподавателей; каталог ссылок на образовательные интернет-ресурсы ( <a href="http://Window.edu.ru">http://Window.edu.ru</a> )
2	Национальная электронная библиотека (НЭБ) — свободный доступ читателей к фондам российских библиотек. Содержит коллекции оцифрованных документов (как открытого доступа, так и ограниченных авторским правом), а также каталог изданий, хранящихся в библиотеках России. ( <a href="http://нэб.рф/">http://нэб.рф/</a> )

## **10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине**



<b>Наименование специальных помещений и помещений для самостоятельной работы</b>
учебные аудитории для проведения учебных занятий
помещения для самостоятельной работы

Материально-техническое обеспечение и организация образовательного процесса по дисциплине для инвалидов и лиц с ограниченными возможностями здоровья осуществляется в соответствии с «Положением об обучении инвалидов и лиц с ограниченными возможностями здоровья».

**ПРИЛОЖЕНИЕ А**  
**ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**  
**ПО ДИСЦИПЛИНЕ «Проектирование пользовательских интерфейсов»**

**1. Перечень оценочных средств для компетенций, формируемых в результате освоения дисциплины**

<b>Код контролируемой компетенции</b>	<b>Способ оценивания</b>	<b>Оценочное средство</b>
ПК-3: Способен проектировать пользовательские интерфейсы по готовому образцу или концепции интерфейса	Экзамен	Комплект контролирующих материалов для экзамена

**2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания**

Оцениваемые компетенции представлены в разделе «Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций» рабочей программы дисциплины «Проектирование пользовательских интерфейсов».

При оценивании сформированности компетенций по дисциплине «Проектирование пользовательских интерфейсов» используется 100-балльная шкала.

<b>Критерий</b>	<b>Оценка по 100-балльной шкале</b>	<b>Оценка по традиционной шкале</b>
Студент освоил изучаемый материал (основной и дополнительный), системно и грамотно излагает его, осуществляет полное и правильное выполнение заданий в соответствии с индикаторами достижения компетенций, способен ответить на дополнительные вопросы.	75-100	<i>Отлично</i>
Студент освоил изучаемый материал, осуществляет выполнение заданий в соответствии с индикаторами достижения компетенций с не принципиальными ошибками.	50-74	<i>Хорошо</i>
Студент демонстрирует освоение только основного материала, при выполнении заданий в соответствии с индикаторами достижения компетенций допускает отдельные ошибки, не способен систематизировать материал и делать выводы.	25-49	<i>Удовлетворительно</i>
Студент не освоил основное содержание изучаемого материала, задания в соответствии с индикаторами	<25	<i>Неудовлетворительно</i>

достижения компетенций не выполнены или выполнены неверно.		
--	--	--

### **3. Типовые контрольные задания или иные материалы, необходимые для оценки уровня достижения компетенций в соответствии с индикаторами**

#### *1. Типовое задания для разработки пользовательского интерфейса*

<b>Компетенция</b>	<b>Индикатор достижения компетенции</b>
ПК-3 Способен проектировать пользовательские интерфейсы по готовому образцу или концепции интерфейса	ПК-3.1 Проектирует интерфейс по концепции или по образцу уже спроектированной части интерфейса

Спроектируйте интерфейс по концептуальной модели заданных задач. Опишите алгоритм решения задачи (блок-схема), тестовый набор данных. Листинг рабочего модуля программы.

- 1) Предметная область «Учет книг». По выбранному из списка автору на экран должен выводиться список книг данного автора. Разработать интерфейс приложения с применением компонентов: ListBox, ComboBox, CheckBox, RadioButton.
- 2) Разработайте модуль для решения задачи: в заданном линейном массиве целых чисел проведите сортировку пузырьком.
- 3) Разработать приложение «Тест дисциплине». Тест должен содержать 5 вопросов. На каждый вопрос должны быть 5 варианта ответов. По результатам теста должна проставляться оценка и выводиться в информационном окне. Разработать интерфейс приложения с применением компонентов: Memo, RadioButton, Bitbtn, GroupBox.
- 4) Разработайте модуль для решения задачи: в заданном линейном массиве целых чисел найти минимальное, максимальное и среднее значения.
- 5) Разработать приложение «Лабораторные работы». Каждая лабораторная работа должна содержать 2 задания. При выборе пункта меню «Задание №1» на экране появляется подменю из двух пунктов: «Условие задания №1» и «Решение задания №1». Разработать интерфейс приложения с применением компонентов: MainMenu, PopupMenu, StringGrid.
- 6) Разработайте модуль для решения задачи: в заданном двумерном массиве целых чисел найти минимальное, максимальное и среднее значения.

#### **4. Файл и/или БТЗ с полным комплектом оценочных материалов прилагается.**

## **7. Лабораторный практикум КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ по дисциплине «Программирование» Лабораторная работа №1**

**Тема:** Начало работы в среде Delphi. Первый проект. Запуск программы. Настройка приложения.

**Цель работы:** научиться создавать проекты, изменять отлаживать и настраивать приложения в среде Delphi

### **Задание:**

Создайте новый проект. Потренируйтесь добавлять и удалять визуальных компонентов на форму. Поработайте со свойствами объектов с помощью инспектора объектов. Добавьте обработчики событий. Запустите программу из интегрированной среды. Выполните пошаговую отладку с инспектированием значений переменных. Потренируйтесь в создании, добавлении и удалении модулей и форм. Выполните настройку опций проекта, а также настройку опций среды программирования. Сохраните проект.

### **Методические указания по выполнению:**

Откройте новый проект можно, выбрав пункт меню File | New Project. Для работы можно создать форму .

Поместите на форму объект TMemo, а затем TEdit так, чтобы он наполовину перекрывал TMemo. Теперь выберите пункт меню Edit | Send to Back, что приведет к перемещению TEdit вглубь формы, за объект TMemo. Это называется изменением Z-порядка компонент. Поместите кнопку TButton в нижнюю часть формы. Теперь растяните Инспектор Объектов так, чтобы свойства Name и Caption были видны одновременно на экране. Теперь измените имя кнопки на Terminate. Текст, который Вы видите на поверхности кнопки - это значение свойства Caption. Напишите обработчик нажатия на кнопку, который будет закрывать приложение.

Для работы со свойствами нажмите клавишу <Shift> и щелкните на TMemo и затем на TListBox. Теперь оба объекта имеют по краям маленькие квадратики, показывающие, что объекты выбраны. Выбрав два или более объектов одновременно, можно выполнить большое число операций над ними, например, передвигать по форме. Затем попробуйте выбрать пункт меню Edit | Size и установить оба поля Ширину(Width) и Высоту(Height) в Grow to Largest. Теперь оба объекта стали одинакового размера.

Затем выберите пункт меню Edit | Align и поставьте в выравнивании по горизонтали значение Center .

Поскольку выбрано два компонента, то содержимое Инспектора Объектов изменится - он будет показывать только те поля, которые являются общими для объектов. Это означает то, что изменения в свойствах повлияют не на один, а на все выбранные объекты.

Рассмотрим изменение свойств объектов на примере свойства Color. Есть три способа изменить его значение в Инспекторе Объектов. Первый - просто напечатать имя цвета (clRed) или номер цвета. Второй путь - нажать на маленькую стрелку справа и выбрать цвет из списка. Третий путь - дважды щелкнуть на поле ввода свойства Color. При этом появится диалог выбора цвета.

Свойство Font работает аналогично свойству Color.

Дважды щелкните на свойство Items объекта ListBox. Появится диалог, в котором Вы можете ввести строки для отображения в ListBox.

Напечатайте несколько слов, по одному на каждой строке, и нажмите кнопку ОК. Текст отобразится в ListBox.

Сохранение программы:

Создать поддиректорию для программы. Лучше всего создать директорию, где будут храниться все Ваши программы и в ней создать поддиректорию для данной конкретной программы.

После создания поддиректории для хранения программы нужно выбрать пункт меню File | Save Project. Сохранить нужно будет два файла. Первый - модуль (unit), над которым Вы работали, второй - главный файл проекта, который "владеет" Вашей программой. Сохраните модуль под именем MAIN.PAS и проект под именем TIPS1.DPR.

Для разработки обработчика нажатия на кнопку перейдите на форму и дважды щелкните мышкой на объект TButton. Вы попадете в окно Редактора, в котором будет фрагмент кода:

```
procedure TForm1.TerminateClick(Sender: TObject);  
begin  
end;
```

Данный код был создан автоматически и будет выполняться всякий раз, когда во время работы программы пользователь нажмет кнопку Terminate. Определение класса в начале файла теперь включает ссылку на метод TerminateClick:

```
TForm1 = class(TForm)  
Edit1: TEdit;  
Memo1: TMemo;  
Terminate: TButton;  
procedure TerminateClick(Sender: TObject);  
private  
{ Private declarations }  
public  
{ Public declarations }  
end;
```

Для того, чтобы приложение закрывалось при нажатии на кнопку необходимо написать код:

```
procedure TForm1.TerminateClick(Sender: TObject);  
begin  
Close;  
end;
```

Справка по простейшим операциям в Delphi

1. Открыть новый проект:

- a. Автоматически при запуски Delphi
- b. На панели быстрых кнопок New|Application / В главном меню File|New|Application

2. Открыть существующий проект выбрать файл проекта .dpr:

- a. На панели быстрых кнопок Open Project / В главном меню File| Open Project
- b. Ctrl+F11

3. Сохранить проект:
  - a. На панели быстрых кнопок Save All /В главном меню File| Save All
4. Вернуть на экран Форму:
  - a. На панели быстрых кнопок View| Form /В главном меню File| View| Forms
  - b. Shift+F12
5. Запустить программу:
  - a. На панели быстрых кнопок щелкнуть на кнопку в виде зеленого треугольника
  - b. В главном меню Run| Run
  - c. F9
6. Остановить программу:
  - a. щелкнуть на кнопку x закрытие окна формы
7. Аварийная остановка программы:
  - a. В главном меню Run| Program Reset
  - b. Ctrl+F2
8. Всплывающая подсказка
  - a. Подвести курсор мыши к элементу интерфейса выждать паузу
9. Переключение с окна на форму или с формы на окно
  - a. F12
10. Отладка программы:
  - a. F4 – выполнение программы до строки , в которой находится курсор
  - b. F5 – задана точка останова
  - c. F7 – пошаговое выполнение программы с заходом в подпрограммы
  - d. F8 – пошаговое выполнение программы без захода в подпрограммы
11. Отменить текущие изменения:
  - a. Ctrl+Z
  - b. В главном меню Edit| Undo
12. Редактирование кода модуля программы:
  - a. Ctrl+C – копирование /Ctrl+X –вырезать /Ctrl+V –восстановить / Ctrl+Y –удалить строку /Delete – удалить фрагмент

## Лабораторная работа №2

**Тема:** Знакомство с основами визуального программирования.

**Цель работы:** Получить навыки работы в инструментальной среде Delphi. Размещение компонентов на форме и задания их свойств

**Задание:**

**Создание графического интерфейса объекта.** Создать приложение, в котором после запуска на форме печатается некоторый текст, например, «Первый проект на языке Delphi».

1. Создать заготовку проекта, для этого свойству *Caption* (заголовок) формы *Form1* присвоим значение «Первый проект». Для этого активизировать форму *Form1* щелчком мыши (чтобы вокруг компонента появились черные квадратики – маркеры для изменения размера). В окне *Object Inspector (Инспектор объектов)* → *Properties (Свойств объекта)* выбрать свойство *Caption* и присвоить ему значение «Первый проект» (написать в ячейке справа от *Caption* Первый проект).

2. Вывести на форму текстовое сообщение. Это можно сделать различными способами:

- С помощью элемента управления *Label* (Метка)
- С помощью элемента управления *Edit* (Одна строка)
- С помощью элемента управления *Memo* (Несколько строк)

3. Разместить на форме метку *Label1* и присвоить свойству *Caption* (Надпись) значение выводимого текстового сообщения:

- Выбрать на Панели инструментов класс управляющих элементов *TLabel* и разместить экземпляр метки *Label1* на форме *Form1*
- Активизировать метку *Label1* щелчком мыши. В окне *Object Inspector (Инспектор объектов)* → *Properties (Свойства объекта)* выбрать свойство *Caption* (заголовок) и присвоить ему значение «Первый проект на языке Delphi».

4. Разместить на форме текстовое поле *Edit1* и присвоить свойству *Text* значение выводимого сообщения (сделать самостоятельно).


**Событийные процедуры.** Любой объект можно связать с набором процедур, исполняемых при наступлении определенных событий, такие процедуры называют *событийными процедурами*

Двойной щелчок мышью по объекту вызывает окно Программного кода (*Unit*) с пустой заготовкой событийной процедуры (метода). Если осуществить двойной щелчок по метке *Memo1*, то появится заготовка событийной процедуры *TForm1.Memo1Change (...)*; (**эта надпись появляется автоматически!**)

```
procedure TForm1.Memo1Change(Sender: TObject);  
begin  
... end;
```

В теле процедуры (между служебными словами *begin* и *end*) должен быть записан метод, т.е. последовательность операторов, которые будут выполняться при наступлении событий (нажатии на *Memo1*)

5. Выбрать на Панели инструментов класс управляющих элементов *TMemo* и разместить экземпляр многострочного редактора *Memo1* на форме *Form1*. В свойстве *Lines* удалить существующую надпись *Memo1*. (щелкнув по многострочному полю *Memo1*, кликнуть напротив свойства *Lines* и удалить там надпись *Memo1*).

6. Активизировать *Memo1* двойным щелчком мыши, тем самым вызвать окно Программного кода с пустой процедурой  *TForm1.Memo1.Change (...)*; (эта надпись появляется **автоматически**)

7. В теле процедуры записать

```
Memo1.Text:='Второй проект на языке Delphi';
```

т.е. при запуске программы, щелкнув по *Memo1*, появится надпись «Второй проект на языке Delphi»

**Использование кнопок.** Обычно, при работе с событийными процедурами используют Кнопки (*TButton*). Логично, что при нажатии на кнопку с определенным именем происходит какое-либо действие. Теперь необходимо решить предыдущую задачу, но с помощью Кнопки. В дальнейшем большинство задач будет решаться при обработке события *OnClick* – двойной щелчок по кнопке.

8. Выбрать на Панели инструментов класс управляющих элементов *Button* и поместить экземпляр кнопки *Button1* на форме *Form1*

9. Выбрать на Панели инструментов класс управляющих элементов *TLabel* и разместить экземпляр метки *Label2* на форме *Form1*

10. Активизировать кнопку *Button1*. В окне Свойств объекта выбрать свойство *Caption* и присвоить ему значение Моя кнопка (щелкнув по кнопке *Button1*, написать в колонке *Properties* Моя Кнопка в ячейке справа от *Caption*).

11. Активизировать *Button1* двойным щелчком мыши, тем самым вызвать окно Программного кода с пустой процедурой *TForm1.Button1Click(Sender: TObject)*; (эта надпись появляется **автоматически**)

12. В теле процедуры записать

```
Label2.Caption:='Третий проект на языке Delphi';
```

т.е. при запуске программы, щелкнув по Моей кнопке, появится надпись на метке *Label2* «Третий проект на языке Delphi»

Выход из программы легко осуществлять, записывая в событийной процедуре оператор *Close*;

**Установка цвета формы и параметров шрифта.** Сделаем внешний вид более привлекательным и для этого изменим свойства объектов, определяющих их внешний вид.

13. Активизировать форму *Form1*. В окне свойства объекта выбрать свойство *Color* (цвет фона) и двойным щелчком открыть диалоговое окно с цветовой палитрой. Выбрать цвет, например желтый.

14. Активизировать метку *Label1*. В окне свойства объекта установить значения *Color* – синий, *Font* – *Arial*, жирный, 26 размер, травяной цвет символов.

15. Поэкспериментировать с объектами *Memo1*, *Edit1*, *Label2*



Для того, чтобы менять параметры шрифта с помощью кнопки при запуске программы необходимо в событийной процедуре указать, например *Label2.Font.Size:=30; Label2.Font.Color:=clWhite*; Т.е. при нажатии на кнопку размер шрифта метки 2 станет равным 30, а цвет - белым.

Для изменения начертания шрифта пишем

- *Label2.Font.Style:=fsbold; {полужирный}*,
- *Label2.Font.Style:=fsitalic; {курсив}*,
- *Label2.Font.Style:=fsunderline; {полужирный}*.

Если необходимо поменять шрифт пишем *Label2.Font.Name:='Arial'; {Arial}* или *Label2.Font.Name:='Times New Roman'; {Times New Roman}*

### **Задание для самостоятельной работы:**

**1.** Создать проект «Вывод сообщений», в котором на форму выводится текстовое сообщение «Первое задание выполнено!» с помощью метки, одной строки и многострочного редактора, а выход из программы осуществляется щелчком по кнопке Выход. Придумать графический интерфейс программе.

**2.** Создать проект «Вывод сообщений в строку», в котором два различных варианта текста выводятся в строку *Edit1* по щелчку по двум кнопкам. Предусмотреть возможность выхода из программы по третьей кнопке.

**3.** Создать проект «Печать на форме», в котором в нескольких метках при нажатии на кнопки тексты будут со следующими параметрами:

- «Times New Roman, 18, курсив, красный»
- «Courier New, 40, полужирный, зеленый»
- «Arial, 22, подчеркнутый, синий»
- «Webdings, 12, подчеркнутый курсив, желтый»
- «Monotype Corsiva, 32, полужирный»

## **Лабораторная работа №3**

**Тема:** расположение объектов и управляющих элементов на форме.

**Цель работы:** приобрести навыки работы с компонентами форм.

**Задание:**

**Расположение объектов на форме.** Местоположение объекта, которое он будет занимать на форме после запуска приложения на выполнение, можно легко изменить.

Для этого достаточно перетащить объект с помощью мыши в любое место экрана. Если необходимо установить точные значения местоположения и размеров объектов, то это можно сделать, установив значения свойств *Left* (расстояние по горизонтали от левого верхнего угла монитора до верхнего левого угла формы), *Top* (расстояние по вертикали от левого верхнего угла монитора до верхнего левого угла формы), *Width* (ширина формы), *Height* (высота формы). Для этого необходимо

активизировать, например Edit. В окне Свойства объекта последовательно присвоить свойствам *Left*, *Top*, *Width*, *Height* значения 32, 104, 225, 30.

**Расположение управляющих элементов на форме.** Расположение на форме и размеры управляющих элементов можно легко изменять с помощью мыши. Для этого необходимо активизировать объект щелчком мыши (он будет выделен восемью квадратиками - маркерами) и перетащить его на новое место или изменить размер элемента.

Точное месторасположение и размеры выделенных элементов управления отображаются двумя парами чисел: *Left*, *Top*, *Width* и *Height*.

1. Активизировать метку *Label1*. В окне свойства объекта последовательно присвоить свойствам *Left*, *Top*, *Width* и *Height* значения 30, 0, 100, 100.

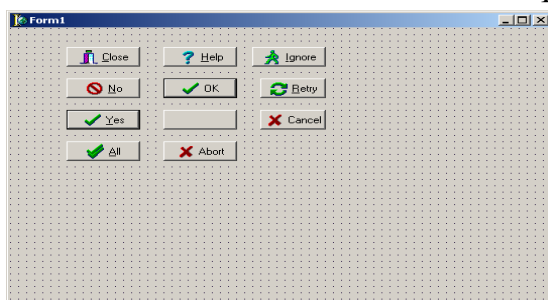
2. Активизировать метку *Label2*. В окне свойства объекта последовательно присвоить свойствам *Left*, *Top*, *Width* и *Height* значения 70, 0, 450, 100

3. Активизировать метку *Label3*. В окне свойства объекта последовательно присвоить свойствам *Left*, *Top*, *Width* и *Height* значения 0, 330, 100, 30.

#### **Задание для самостоятельной работы:**

1. Создать проект «Перемещение кнопки по форме», в котором, например, в центре, располагаются кнопки и по щелчку перемещаются в заданное место на форме, например по углам формы

2. Создать проект «Перемещение формы на экране», в котором будет изменяться местоположение формы на экране монитора с помощью четырех командных кнопок. Кнопки расположить в углах формы. После щелчка по кнопке, форма должна переместиться в соответствующую часть экрана. Перемещение формы должно сопровождаться комментариями, выводимыми с помощью надписи в центре формы.



#### **Кнопка с изображением BitBtn**

Одной из разновидностью кнопок служит *BitBtn* – кнопка с изображением. Она находится во вкладке *Additional*. Свойство *Kind* для такой кнопки определит одну из 11 стандартных разновидностей кнопки. Если у кнопки был изменен рисунок (свойство *Glyph*), *Delphi*

автоматически присваивает свойству *Kind* кнопки значение *bkCustom*.

## 8. Методические рекомендации студентам по изучению дисциплины

Планирование и организация времени, отведенного на изучение дисциплины.

В начале семестра студент знакомится с содержанием и структурой дисциплины. Студент самостоятельно планирует свое время, опираясь на календарный график, приведенный в рабочей программе дисциплины. Все виды работ можно разделить на две группы – контактная работа и самостоятельная работа. Контактная работа может быть аудиторной, внеаудиторной, а также проводиться в электронной информационно–образовательной среде.

Контактная работа при проведении учебных занятий по дисциплине (модулю) включает в себя лекции, лабораторные работы, консультации по выполнению лабораторных работ. Консультации могут быть групповыми или индивидуальными. Контактная работа студентов по дисциплине также может содержать элементы самостоятельной работы. В этом случае она выполняется на учебных занятиях под непосредственным руководством преподавателя и по его заданию. Объем времени на контактную работу студентов регламентируется расписанием занятий.

Самостоятельная работа студентов – планируемая учебная, научно-исследовательская работа студентов, выполняемая во внеаудиторное время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия и не регламентируется расписанием занятий.

Самостоятельная работа, которую студент организует по своему усмотрению, без непосредственного контроля со стороны преподавателя - подготовка к лекциям, практическим занятиям, контрольным работам, экзамену. В этой связи стоит подчеркнуть, что очень важно умение оптимизировать процесс сочетания этих двух частей, необходимо равномерно распределять силы по всей дистанции семестра.

Для успешного освоения материала и качественного выполнения лабораторной работы необходимо после лекции и перед лабораторной работой, повторить материал (15 – 30 минут).

Перед контрольной работой необходимо не только повторить материал по конспекту лекций, но и изучить рекомендуемую литературу по соответствующим темам.

Сценарий изучения дисциплины (последовательность действий):

1. Посещение лекций (регламентируется расписанием занятий).
  2. Выполнение лабораторных работ и выполнение индивидуальных заданий (регламентируется расписанием занятий).
  - 3 Самостоятельная внеаудиторная работа с конспектом лекций и рекомендуемой литературой, решение задач.
  4. Выполнение контрольных работ.
  5. Подготовка к зачету.
- Рекомендации по работе с литературой.

Работа с литературой является основным методом самостоятельного овладения знаниями. Это сложный процесс, требующий выработки определенных навыков, поэтому студенту нужно обязательно научиться работать с книгой.

Осмысление литературы требует системного подхода к освоению материала. В работе с литературой системный подход предусматривает не только тщательное (при необходимости – многократное) чтение текста и изучение специальной литературы, но и обращение к дополнительным источникам – справочникам, энциклопедиям, словарям. Эти источники – важное подспорье в самостоятельной работе студента, поскольку глубокое изучение именно их материалов позволит студенту уверенно «распознавать», а затем самостоятельно оперировать теоретическими категориями и понятиями, следовательно – освоить новейшую научную терминологию. Такого рода работа с литературой обеспечивает решение студентом поставленной перед ним задачи (подготовка к практическому занятию, выполнение контрольной работы и т.д.).

Выбор литературы для изучения делается обычно по предварительному списку литературы, который выдал преподаватель, либо путем самостоятельного отбора материалов. После этого непосредственно начинается изучение материала, изложенного в книге.

Прежде чем приступить к чтению, необходимо запомнить или записать выходные данные издания: автор, название, издательство, год издания, название интересующих глав. Предисловие или введение книги поможет установить, на кого рассчитана данная публикация, какие задачи ставил перед собой автор. Это помогает составить представление о степени достоверности или научности данной книги. Содержание (оглавление) дает представление о системе изложения ключевых положений всей публикации и помогает найти нужные сведения. Если в книге есть главы или отдельные параграфы, которые соответствуют исследуемой теме дисциплины, то после этого необходимо ознакомиться с введением.

Во введении или предисловии разъясняются цели издания, его значение, содержится краткая информация о содержании глав работы. Иногда полезно после этого посмотреть послесловие или заключение. Особенно это важно, если это не учебник, а монография, потому что в заключении объясняется то, что может оказаться непонятным при изучении материала. В целом, это поможет правильно структурировать полученные знания.

После просмотра книги целиком или отдельной главы, которая была необходима для изучения определенной темы курса, нужно сделать записи в виде краткого резюме источника. В таком резюме следует отразить основную мысль изученного материала, приведенные в ее подтверждение автором аргументы, ценность данных аргументов и т.п. Данные аргументы помогут сформировать собственную оценку изучаемого вопроса.

Во время изучения литературы необходимо конспектировать и составлять рабочие записи прочитанного. Такие записи удлиняют процесс проработки, изучения книги, но способствуют ее лучшему осмыслению и усвоению, выработке навыков кратко и точно излагать материал. В идеале каждая подобная запись должна быть сделана в виде самостоятельных ответов на вопросы, которые задаются в конце

параграфов и глав изучаемой книги. Однако такие записи могут быть сделаны и в виде простого и развернутого плана, цитирования, тезисов, резюме, аннотации, конспекта.

При изучении литературы особое внимание следует обращать на новые термины и понятия. Понимание сущности и значения терминов способствует формированию способности логического мышления, приучает мыслить абстракциями, что важно при усвоении дисциплины. Поэтому при изучении темы курса студенту следует активно использовать универсальные и специализированные энциклопедии, словари, иную справочную литературу.

Вся рекомендуемая для изучения курса литература подразделяется на основную и дополнительную. К основной литературе относятся источники, необходимые для полного и твердого усвоения учебного материала (учебники и учебные пособия). Необходимость изучения дополнительной литературы диктуется прежде всего тем, что в учебной литературе (учебниках) зачастую остаются неосвещенными современные проблемы, а также не находят отражение новые документы, события, явления, научные открытия последних лет. Поэтому дополнительная литература рекомендуется для более углубленного изучения программного материала.

### Список литературы

1. Дудник, Е.А. Программирование. Часть III (лабораторный практикум по программированию в среде Delphi) [текст]: учебно-метод. пособие для студентов направления ИВТ/ Е.А. Дудник, Н.С. Зорина. - Рубцовск: РИИ, 2017. - 63 с. (6 экз.+ЭР)

2. Культин Н.Б. Delphi в задачах и примерах : [текст]. -3-е изд., испр. и доп./ СПб.: БХВ-Петербург, 2012 - 288 с. (10 экз.)

3. Мандел, Т. Разработка пользовательского интерфейса / Т. Мандел. — Москва : ДМК Пресс, 2007. — 418 с. — ISBN 5-94074-069-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/1227> (дата обращения: 18.01.2021). — Режим доступа: для авториз. пользователей.

4. Стивенс, Р. Delphi. Готовые алгоритмы : учебное пособие / Р. Стивенс. — Москва : ДМК Пресс, 2007. — 384 с. — ISBN 5-94074-106-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/1234> (дата обращения: 05.02.2021). — Режим доступа: для авториз. пользователей.

5. Рояк, М.Э. Программирование под Windows графических интерфейсов пользователя : учебное пособие : [16+] / М.Э. Рояк, И.М. Ступаков ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2018. – 72 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=575018> (дата обращения: 15.11.2020). – Библиогр. в кн. – ISBN 978-5-7782-3754-4. – Текст : электронный.

6. Белов, В. В. Программирование в Delphi: процедурное,

объектно-ориентированное, визуальное : учебное пособие / В. В. Белов, В. И. Чистякова. — 2-е изд., стер. — Москва : Горячая линия-Телеком, 2017. — 240 с. — ISBN 978-5-9912-0412-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/111017> (дата обращения: 05.02.2021). — Режим доступа: для авториз. пользователей.

7. [https://www.studmed.ru/mandel-t-razrabotka-polzovatelskogo-interfeysa\\_1b011418471.html](https://www.studmed.ru/mandel-t-razrabotka-polzovatelskogo-interfeysa_1b011418471.html)

8. [www.books.ru](http://www.books.ru) / Купер А., Рейман Р., Кронин Д. Алан Купер об интерфейсе. Основы проектирования взаимодействия. — Пер. с англ. — СПб.: Символ'Плюс, 2009 — 688 с..