

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Шебзузова Татьяна Александровна

Должность: Директор Пятигорского института (филиал) Северо-Кавказского
Федерального государственного автономного образовательного учреждения высшего образования
федерального университета

Дата подписания: 06.09.2023 12:54:28

Уникальный программный ключ:

d74ce93cd40e39275c3ba2f58486412a1e8e1981

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Пятигорский институт (филиал) СКФУ

Колледж Пятигорского института (филиал) СКФУ

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

Специальности СПО

09.02.07 Информационные системы и программирование

Пятигорск

Методические указания для практических занятий по дисциплине ОП.04 Основы алгоритмизации и программирования составлены в соответствии с требованиями ФГОС СПО. Предназначены для студентов, обучающихся по специальности 09.02.07 Информационные системы и программирование.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Программа Основы алгоритмизации и программирования предусматривает изучение алгоритмизации как базовой составляющей технологического процесса создания программного продукта, а так же изучение языка программирования и формирование навыков разработки ИС, объектно-ориентированного программирования у студентов.

При изучении предмета следует соблюдать единство терминологии и обозначения в соответствии с действующими стандартами, Международной системой единицы (СИ).

В результате изучения Основ алгоритмизации и программирования студенты *должны уметь:*

- создавать проект по разработке приложения и формулировать его задачи;
- использовать языки структурного, объектно-ориентированного программирования и языка сценариев для создания независимых программ;
- разрабатывать графический интерфейс приложения;
- использовать методы тестирования в соответствии с техническим заданием.

знать:

- национальную и международную систему стандартизации и сертификации и систему обеспечения качества продукции;
- методы контроля качества объектно-ориентированного программирования, файлового ввода-вывода, создание сетевого сервера и сетевого клиента;
- платформы для создания исполнения и управления информационной системой;
- особенности программных средств, используемых при разработке ИС.

В результате освоения учебной дисциплины студент должен овладевать:

Общими компетенциями:

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.

ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.

ОК 09. Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке.

Профессиональными компетенциями:

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

ОБЩИЕ МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

По Основам алгоритмизации и программирования практические занятия содержат задачи и теоретические вопросы. Варианты для каждого обучающегося - индивидуальные.

Задачи и ответы на вопросы, выполненные не по своему варианту, не засчитываются.

Практическое занятие выполняется в отдельной тетради. Условия задачи и формулировки вопросов переписываются полностью. Формулы, расчеты, ответы на вопросы пишутся ручкой, а чертежи, схемы и рисунки выполняются карандашом, на графиках и диаграммах указывается масштаб. Вначале задача решается в общем виде, затем делаются расчёты по условию задания. Решение задач обязательно ведется в Международной системе единиц (СИ).

При выполнении практического занятия необходимо следовать методическим указаниям: повторить краткое содержание теории, запомнить основные формулы и законы, проанализировать пример выполнения аналогичного задания, затем преступить непосредственно к решению задачи. К зачету допускаются студенты, получившие положительные оценки по всем практическим работам.

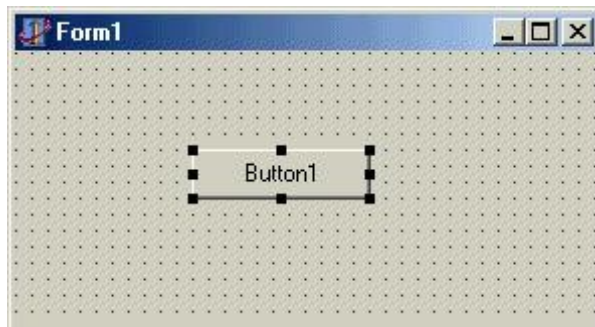
Правила выполнения практических занятий.

1. Студент должен прийти на практическое занятие подготовленным к выполнению практической работы.
2. Каждый студент после проведения работы должен представить отчет о проделанной работе с анализом полученных результатов и выводом по работе.
3. Таблицы и рисунки следует выполнять с помощью чертежных инструментов (линейки, циркуля, и т.д.) карандашом с соблюдением ЕСКД.
4. Расчет следует проводить с точностью до двух значащих цифр.
5. Исправления проводить на обратной стороне листа. При мелких исправлениях неправильное слово (буква, число и т.п.) аккуратно зачеркивается и над ним пишут правильное пропущенное слово (букву, число и т.п.).
6. Вспомогательные расчеты можно выполнять на отдельных листах, а при необходимости на листах отчета.
7. Если студент не выполнит практическую работу или часть работы, то он выполнит ее во внеурочное время, согласованное с преподавателем.
8. Оценку по практической работе студент получает с учетом срока выполнения работы, если;
 - расчеты выполнены правильно и в полном объеме;
 - сделан анализ проделанной работы и вывод по результатам работы;
 - студент может пояснить выполнение любого этапа работы;
 - отчет выполнен в соответствии с требованиями к выполнению работы.

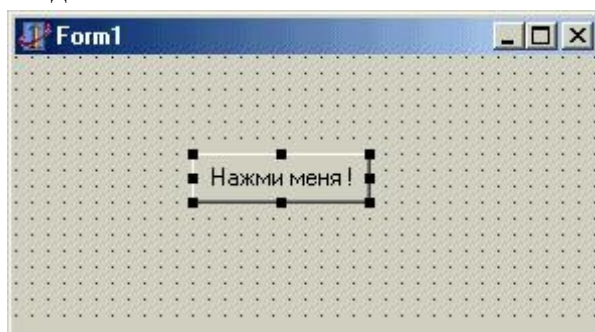
ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №1. СРЕДА ПРОГРАММИРОВАНИЯ DELPHI. ПЕРВАЯ ПРОГРАММА.

1. Запустите Delphi
2. Сохраните программу в папке «Первый проект». Дайте файлу проекта (расширение “**dpr**”) название Hello и файлу модулю формы (расширение “**pas**”) дайте произвольное название на английском языке.

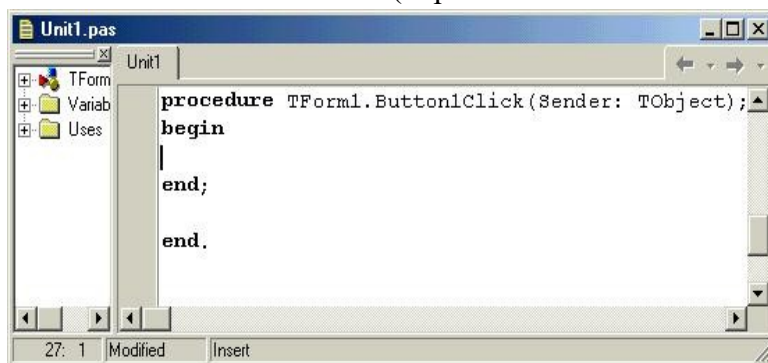
3. Установите на форму компонент **Button**  так, как показано на рисунке.



4. Затем для кнопки в инспекторе объектов в свойстве **Caption** данной кнопки впечатайте текст «**Нажми меня**» и надпись на кнопке соответственно изменится:



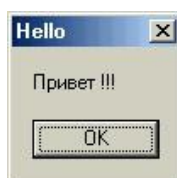
5. Теперь щелкните дважды мышью по кнопке. Вы попадете в редактор кода, где увидите текст, автоматически созданный **Delphi**. Созданная вами процедура называется обработчиком события нажатия кнопки Button1 (обработчик события OnClick).



6. Впечатайте там, где находится текстовый курсор (между **begin** и **end** процедуры)

строку: **ShowMessage('Привет !!!');**

7. После этого запустите проект на выполнение, нажав клавишу **F9**. На экране появится окно созданной нами программы. Щелкните по кнопке и увидите запрограммированный Вами привет:



Итак, Вы создали свою первую программу.

Для работы программы достаточно только, чтобы на компьютере была установлена ОС **Windows**. Если Вы заглянете при помощи проводника в папку **Первый проект**, то увидите там файл **Hello.exe**. Этот файл можно переписать на другой компьютер, и он будет работать, даже если на этом компьютере не установлена **Delphi**.

Но если мы захотим совершенствовать нашу первую программу, то понадобятся, кроме **Delphi**, и другие файлы, которые можно увидеть в папке **Hello** – о их назначении вы узнаете позже.

Практическое задание: в созданный вами проект добавьте еще три кнопки. Каждая кнопка должна выдавать разные сообщения: сообщение об ошибке, предупреждающее сообщение и информационное окно. Придумайте соответствующий текст. Каждое окно сообщения должно содержать разные наборы кнопок.

ДОПОЛНИТЕЛЬНАЯ ТЕОРИЯ Как выдать сообщение по нажатию на кнопку:

1 способ: в обработчике события нажатия кнопки **OnClick** вписать строку **ShowMessage('Текст');**.

Вместо слова **Текст** может быть вписан произвольный текст. В результате по нажатию на кнопку будет выдаваться небольшое окно с кнопкой «ОК», которое будет содержать **Текст**, введенный вами.

2 способ: в обработчике события нажатия кнопки **OnClick** вписать, например следующую строку:

MessageDlg('Текст', mtInformation, mbOKCancel, 0); В результате по нажатию на кнопку будет выдаваться следующее окно:



Можно управлять внешним видом этого окна. Для чего необходимо настраивать его параметры. **Параметры функции MessageDlg.**

1. Текст, который будет выводиться в окне.
2. Определяет тип окна (и значок). Может быть одним из значений:

Возможные значения параметра	Описание	Значок, появляющийся в окне
mtInformation	информационное окно	
mtWarning	окно предупреждения	
mtError	окно, сообщающее об ошибке	

mtConfirmation	вопросительное окно	  
mtCustom	простое окно без значка	

3. Определяет, какие кнопки будут расположены в окне. Этот параметр может иметь одно из следующих значений:

- mbYesNoCancel
- mbYesAllNoAllCancel
- mbOKCancel
- mbAbortRetryIgnore
- mbAbortIgnore

Таким образом, можно объявлять наборы кнопок. Если вы хотите, чтобы в окне сообщения была только одна кнопка (например, ОК), либо произвольный набор кнопок, то их нужно указывать в квадратных скобках.

Например:

- [mbOK]
- [mbOK, mbAbort]

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №2. ЛИНЕЙНЫЕ ПРОСТЕЙШИЕ ПРОГРАММЫ

Уровень №1. Даны три числа x, y, z . Составить программу для вычисления значения переменных a и b по следующим формулам:

$a = \frac{1 + \sin^2(x + y)}{2 + \sqrt{x - 1 + x^2 y^2}} + x$	$b = \cos^2\left(\arctg \frac{1}{z}\right)$
--	---

Решение:

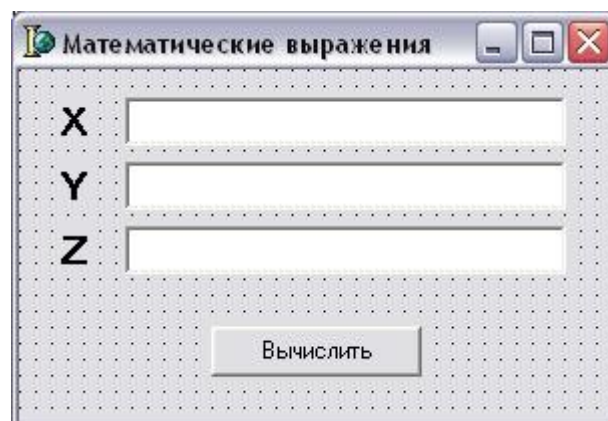
1. Запустите **Delphi**. Перед вами откроется новый проект. Сохраните его в отдельной папке (сохраните модуль **.pas** и файл проекта **.dpr**) командой из главного меню **File – Save All**.
2. Установите на форму 3 компонента **Edit** (со вкладки **Standart** на палитре компонентов), 3 компонента **Label** (стр. **Standart**), кнопку **Button** (стр. **Standart**).
3. Установите в окне **Object Inspector** свойства следующих компонентов:
- 4.

Компонент	Свойство	Значение
Форма	Caption	Математические выражения
	Name	Form1

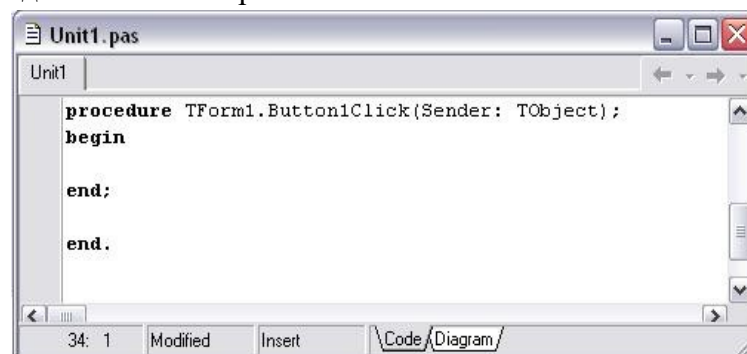
Label1	Caption	X
Label2	Caption	Y
Label3	Caption	Z
Edit1	Text	
Edit2	Text	
Edit3	Text	
Button1	Caption	Вычислить

Свойство **Font** у всех компонентов установите: Начертание - Жирный, Размер - 14 пунктов.

4. В результате ваша форма должна иметь следующий вид:



5. Затем дважды щелкните левой кнопкой мыши по кнопке **Button1**. В результате откроется окно редактора кода с готовым обработчиком события onClick:



6. Впишите между служебными словами **begin** и **end** следующий код программы:

```

x:=StrToInt(Edit1.Text);
y:=StrToInt(Edit2.Text);
z:=StrToInt(Edit3.Text);          a:=(1+sqr(sin(x+y)))/(2+abs(x-2*x/(1+sqr(x)*sqr(y))))+x;
b:=sqr(cos(arctan(1/z)));
ShowMessage('a='+FloatToStr(a)+' , b='+FloatToStr(b));

```

7. Перед словом **begin** добавьте раздел **Var**: **var x,y,z:integer;a,b:real;**

8. Сохраните проект (**File – Save All**).

9. Запустите программу на выполнение (**F9** или **Run - Run**).

10. Введите экспериментальные данные в компоненты **Edit1**, **Edit2**, **Edit3** и нажмите на кнопку «Вычислить». В результате на экране появиться окно с вычисленными значениями переменных a и b.

Уровень №2. Даны три числа x,y,z. Составить программу для вычисления значения переменных a и b по следующим формулам:

$a = \frac{2 \cos(x - \frac{\pi}{6})}{1/2 + \sin^2 y}$	$b = 1 + \frac{z^2}{2} + \frac{z}{5}$
--	---------------------------------------

Уровень №3. Даны два числа a и b. Получить их сумму, разность и произведение.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №3. РЕШЕНИЕ ЗАДАЧ ПО ЛИНЕЙНЫМ ПРОСТЕЙШИМ ПРОГРАММАМ

Уровень №1. Составьте программу, которая переводит введенную пользователем сумму в тенге в доллары.

— Тест: сумма в тенге = 2500, курс доллара = 130, результат = 19,23

Уровень №2

1. Пользователь вводит цену 1 килограмма яблок и вес в килограммах своей покупки. Программа выводит общую стоимость.

Тест: цена = 130, вес = 3, результат = 390.

2. Даны действительные числа x и y. Получить $z = (|x| - |y|) / (1 + |x \cdot y|)$.

— Тест: x=7, y=5, результат z=0.55

Уровень №3. Даны катеты прямоугольного треугольника. Найти его гипотенузу и площадь. (Формула гипотенузы - $c^2 = a^2 + b^2$, Формула площади - $S = ab/2$).

— Тест: a=3, b=4, результат c=5, s=6

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №4. УСЛОВНЫЙ ОПЕРАТОР IF

Уровень №1:

Составить программу, нахождения наибольшего из двух введенных чисел на экран.

— Тест: x=5, y=7, результат y – максимальный.

Уровень №2:

$$\begin{cases} x^2, & \text{если } x \leq 0 \\ - \end{cases}$$

$$y = \begin{cases} x, & \text{если } x > 0 \end{cases}$$

1. Составить программу вычисления значения функции:

Тест 1: $x=25$, результат $y=5$.
Тест 2: $x=-2$, результат $y=4$.
Тест 3: $x=0$, результат $y=0$

2. Если значение переменной X отрицательное, то поменять его на положительное, иначе оставить без изменения.

Тест 1: $x = -5$, результат $y=5$

Тест 2: $x = 5$, результат $y=5$

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №5. ПРИМЕНЕНИЕ ОПЕРАТОРА IF

Уровень №3

1. Даны три числа. Написать программу для подсчета количества чисел равных 0.

Тест 1: $x=0, y=2, z=3$, результат $k=1$

Тест 2: $x=0, y=0, z=0$, результат $k=3$

Тест 3: $x=1, y=3, z=4$, результат $k=0$

2. Составить программу, которая проверяет, делится ли на 13 целое число, введенное пользователем с клавиатуры.

Тест 1: $x=26$, результат 'делится нацело'

Тест 2: $x=25$, результат 'не делится на цело'

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №6. ВЛОЖЕННЫЕ УСЛОВНЫЕ ОПЕРАТОРЫ

Уровень №1.

Даны действительные числа a, b, c ($a > 0$). Полностью исследовать квадратное уравнение $ax^2 + bx + c = 0$, т.е. если действительных корней нет, то должно быть выдано сообщение об этом, иначе должны быть выданы один или два корня.

Тест 1: $a=1, b=2, c=3$, результат – корней нет.

Тест 2: $a=2, b=6, c=3$, результат – $x_1 = -0,211324865405187, x_2 = -0,788675134594813$.

Тест 3: $a=1, b=2, c=1$, результат – $x = -1$.

Уровень №2

Составить программу вычисления стоимости покупки с учетом скидки. Скидка в 3% предоставляется в том случае, если сумма покупки больше 5000 тенге, в 5% - если сумма

больше 10000 тенге.

Тест 1: $x=10001$, результат = 9500,95.

Тест 2: $x=5001$, результат = 4850,97.

Тест 3: $x=4900$, результат = 4900.

Уровень №3.

Составить программу нахождения произведения двух наибольших из трех вводимых с клавиатуры.

Тест 1: $x=1, y=2, z=3$, результат – 6

Тест 2: $x=3, y=2, z=4$, результат – 12

Тест 3: $x=3, y=2, z=1$, результат – 6

Тест 4: $x=3, y=1, z=2$, результат – 6

Тест 5: $x=1, y=3, z=2$, результат – 6

Тест 6: $x=4, y=5, z=2$, результат – 20

Тест 7: $x=1, y=1, z=1$, результат – все числа одинаковые

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №7.

ПРИМЕНЕНИЕ ВЛОЖЕННЫХ УСЛОВНЫХ ОПЕРАТОРОВ

Уровень №1.

Написать программу, для вычисления значения функции

$$y = \begin{cases} x^2, & \text{при } x > 2 \\ x + 1.5, & \text{при } x = 2 \\ \sqrt{x} & \text{при } 2 > x \geq 0 \end{cases}$$

Тест 1: $x = 2$, результат: $y = 3.5$

Тест 2: $x = 3$, результат $y=9$

Тест 3: $x=0$, результат $y=0$

Уровень №2.

Даны действительные числа x, y, z . Получить $\text{Min}(x, y, z)$.

Тест 1. $x=1, y=2, z=3$, результат = 1

Тест 2. $x=2, y=3, z=1$, результат = 1

Тест 3. $x=2, y=3, z=0$, результат = 0

Уровень №3.

Составьте программу, которая вычисляет дату следующего дня. Вводить числа даты, месяца и года в три разных Edit и заносить в три разные переменные, например: day, month, year.

Тест: дата сегодня – 5 10 2005, результат – 6.10.2005

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №8.

РЕШЕНИЕ ЗАДАЧ ПО ВЛОЖЕННЫМ УСЛОВНЫМ ОПЕРАТОРАМ

Уровень №1.

Пользователь вводит целые числа x , y , z . Если первое число самое большое, найти сумму квадратов введенных чисел, если наибольшее – второе число – найти разность квадратов введенных чисел, иначе частное от деления произведения всех чисел на их сумму.

Тест 1. $x=4, y=2, z=3$, результат = 29

Тест 2. $x=2, y=3, z=1$, результат = -6

Тест 3. $x=2, y=3, z=5$, результат = 3

Уровень №2.

Заданы действительные числа a , b , c , d . Найти полусумму наибольшего и наименьшего из них.

Тест 1. $a=4, b=2, c=3, d=1$, результат = 2,5

Тест 2. $a=1, b=0, c=2, d=1$, результат = 1

Уровень №3.

Составьте программу, которая «задумывает» случайное число в промежутке от 1 до 10 и предлагает пользователю его отгадать с трех попыток. При этом, если вводимое пользователем число больше загаданного, то программа печатает слово «Больше!», а если меньше, то слово «Меньше!».

$Y:=\text{Random}(9)+1$; {загадываем случайное число в диапазоне от 1 до 10} $X:=\text{StrToInt}(\text{InputBox}(\text{'Ввод числа'}, \text{'Введите число'}, ''))$;

If $x=y$ then Memo1.lines.add('вы угадали')

Else

Begin

If $x>y$ then Memo1.Lines.Add('Меньше')

Else Memo1.Lines.Add('Больше');

$X:=\text{StrToInt}(\text{InputBox}(\text{'Ввод числа'}, \text{'Введите число'}, ''))$;

... {Дописать самостоятельно еще 2 попытки} End;

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №9.

ОПЕРАТОР ВЫБОРА CASE

$$y = \begin{cases} x+1, & \text{при } b=3 \\ 2x-3, & \text{при } b=2,4,6 \\ e^x+5, & \text{при } b=7,8,9 \end{cases}$$

—
 x^2 , при $b \neq 2,3,4,6,7,8,9$

Уровень №1.

Вычислить следующую функцию:

Тест 1. $b=3, x=3$, Результат $y=4$

Тест 2. $b=4, x=4$, Результат $y=5$

Уровень №2.

Составить программу, которая после введенного с клавиатуры числа (в диапазоне от 1 до 99), обозначающего денежную единицу, дописывает слово «копейка» в правильной форме. Например: 5 копеек, 41 копейка и т.д.

Тест 1. $x=5$, Результат – 5 копеек

Тест 2. $x=11$, результат – 11 копеек

Тест 3. $x=1$, результат – 1 копейка

Тест 4. $x=12$, результат – 12 копеек

Тест 5. $x=2$, результат – 2 копейки

Тест 6. $x=51$, результат – 51 копейка

Тест 7. $x=99$, результат – 99 копеек

Уровень №3.

Составить программу, которая после введенного с клавиатуры числа (в диапазоне от 1 до 999), обозначающего денежную единицу, дописывает слово «рубль» в правильной форме. Например: 12 рублей, 21 рубль и т.д.

Тест 1. $x=5$, Результат – 5 рублей

Тест 2. $x=11$, результат – 11 рублей

Тест 3. $x=1$, результат – 1 рубль

Тест 4. $x=12$, результат – 12 рублей

Тест 5. $x=2$, результат – 2 рубля

Тест 6. $x=51$, результат – 51 рубль

Тест 7. $x=99$, результат – 99 рублей

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №10.

ЦИКЛ С ПАРАМЕТРОМ

Уровень №1.

Найти сумму положительных нечетных чисел, меньших 100.

Тест: результат = 2500.

Уровень №2

1. Найти сумму целых положительных чисел из промежутка от А до В, кратных четырех. Значения переменных А и В вводятся пользователем.

Тест: $A=15$, $B=35$, результат = 120.

2. Найти сумму целых положительных чисел, больших 20, меньших 100, кратных 3 и заканчивающихся на 2, 4 или 8.

Тест: результат = 402.

Уровень №3.

Написать программу поиска двухзначных чисел, обладающих следующим свойством: если к сумме цифр числа прибавить квадрат этой суммы, то получится снова данное число.

Тест: результат – числа 12, 42, 90.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №11. ЦИКЛ С ПРЕДУСЛОВИЕМ И С ПОСТУСЛОВИЕМ

Уровень №1. Вводится натуральное число. Определить сумму квадратов его цифр.

Тест 1: $x=432$, результат – 29

Тест 2: $x=1234$, результат - 30

Уровень №2.

1. Вводится натуральное число. Определить, количество цифр 5 в нем. Тест 1: $x=1525$, результат – 2 Тест 2: $x=25$, результат - 1

2. Найти первое число Фибоначчи, большее m ($m>1$).

Уровень №3.

Вычислить сумму всех чисел Фибоначчи, не превосходящих 1000. Тест: результат – 2583

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №12. ВЛОЖЕННЫЕ ЦИКЛЫ

Уровень №1.

Даны натуральные числа n и k . Составить программу для вычисления выражения $1^k \square 2^k \square \dots \square n^k$.

Тест 1: $n=3, k=1$, результат 6

Тест 2: $n=3, k=2$, результат 14

Уровень №2. Дано натуральное число n . Составить программу для вычисления выражения $1^1 \square 2^2 \square \dots \square n^n$.

Тест 1: $n=3$, результат 32

Уровень №3. Старинная задача. Сколько можно купить быков, коров и телят, если плата за быка – 10 рублей, за корову – 5 рублей, за теленка – полтинник (0,5 рубля) и на 100 рублей надо купить 100 голов скота?

Тест: быков 1, коров 9, телят 90

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №13. ОДНОМЕРНЫЕ МАССИВЫ

Уровень №1.

Подсчитать число и сумму положительных, число и произведение отрицательных элементов заданного массива A(5).

Тест: A(1, -2, 3, -8, 7), результат: количество положительных = 3, сумма положительных = 11, количество отрицательных = 2, произведение отрицательных = 16.

Уровень №2.

1. Вычислить сумму квадратов всех элементов заданного массива X(4), за исключением элементов, кратных пяти.

Тест 1: X(1, 5, 4, 2), результат = 21

Тест 2: X(5, 5, 5, 5), результат = 0

2. Элементы заданного массива B(5) переписать в новый массив A(5) в обратном порядке.

3.

Тест: B(1, 2, 3, 4, 5), результат A(5, 4, 3, 2, 1)

Уровень №3.

Известно, что в массиве A(8) имеются элементы, равные 5. определить:

а номер первого из них;

б номер последнего из них;

Тест: A(1, 5, 4, 3, 5, 6, 5, 8), результат: номер первого = 2, номер последнего = 7

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №14. ДВУМЕРНЫЕ МАССИВЫ

Уровень №1.

1. Установите на форму компонент StringGrid со страницы Additional.
2. Установите количество столбцов (свойство ColCount)=6, количество строк (RowCount)=6.
3. Установите ширину столбца (DefaultColWidth)=24, высоту строки (DefaultRowHeight)=24.
4. Установите цвет компонента (Color)=clGreen, а цвет фиксированных элементов (FixedColor)=clMaroon.
5. В свойстве Options установите в значение True подсвойства GoEditing и GoTabs.
6. Запустите программу и внесите экспериментальные данные в ячейки таблицы. Перемещаться между ячейками таблицы можно с помощью клавиши Tab, расположенной на клавиатуре. В результате должен получиться примерно следующий вид таблицы:

	4	5	8	8	9
	4	5	7	9	3
	5	7	9	23	3
	3	1	3	7	8
	5	7	8	9	0

7. Закройте выполняемую программу и вернитесь в Delphi.

8. Установите на форму кнопку и в обработчике события нажатия кнопки внесите следующий код программы:

```
var i,j,k:integer; begin
  k:=0;
  for i:=1 to 5 do for j:=1 to 5 do begin
    k:=k+1;
    StringGrid1.Cells[j,i]:=inttostr(k);
  end;
end;
```

9. После запуска программы нажмите на кнопку и в результате таблица должна принять следующий вид:

	1	2	3	4	5
	6	7	8	9	10
	11	12	13	14	15
	16	17	18	19	20
	21	22	23	24	25

Уровень №2.

1. Дана матрица $A(N,M)$. Найти её наименьший элемент и номера строки и столбца, на пересечении которых он находится.
2. Дан двумерный массив. Верно ли, что сумма элементов главной диагонали массива не превышает 100?

Уровень №3.

Дана матрица A размерности 5×5 . Найти K элементов побочной диагонали и обнулить элементы главной диагонали. Результат вывести на экран.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №15. ТИПОВЫЕ АЛГОРИТМЫ ОБРАБОТКИ ОДНОМЕРНЫХ МАССИВОВ

Уровень №1. Дан массив A , состоящий из 10 элементов. Найти максимальный и минимальный элементы массива.

Тест: $A(1,3,4,5,10,7,6,2,3,0)$, результат – максимальный = 10, минимальный = 0.

Уровень №2.

1. Дан массив $A(5)$. Определить, имеется ли между минимальным и максимальным элементами этой последовательности хотя бы одно отрицательное значение.

Тест 1: $A(1, -3, -2, 5, 10)$, результат – Да.

Тест 2: $A(1, 2, 3, 4, 5)$, результат – Нет.

2. Дан массив $A(5)$. Отрицательные элементы заменить на положительные (взять по модулю), нулевые значения заменить числом ПИ и результат вывести на экран.

Тест: A(1, -5, 2, 0, -9), результат - A(1, 5, 2, 3.14, 9).

Уровень №3. Дан массив A(5), вещественных чисел. Выяснить:

а верно ли, что максимальный элемент превышает минимальный на 25; б верно ли, что минимальный элемент меньше максимального в два раза? Тест 1: A(1, 26, 2, 4, 6), результат – пункт а верно, пункт б не верно Тест 2: A(2, 4, 3, 3, 3), результат – пункт а не верно, пункт б верно.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №16. МЕТОДЫ СОРТИРОВКИ ДАННЫХ В МАССИВЕ

Уровень №1. Определить, является ли массив упорядоченным по возрастанию. В случае отрицательного ответа определить номер первого элемента, «нарушающего» такую упорядоченность.

Уровень №2.

1. В массиве должна быть записана информация о результатах соревнований по плаванию, в котором приняли участие 25 спортсменов. Данные должны быть записаны в порядке ухудшения результата. После заполнения массива выяснилось, что значение первого элемента не соответствует требованию упорядоченности. Изменить массив так, чтобы данные были упорядочены.

2. В массиве должна быть записана информация о количестве жителей в каждом из 30 городов (в порядке возрастания численности). После заполнения массива выяснилось, что значение последнего элемента не соответствует требованию упорядоченности. Изменить массив так, чтобы данные были упорядочены.

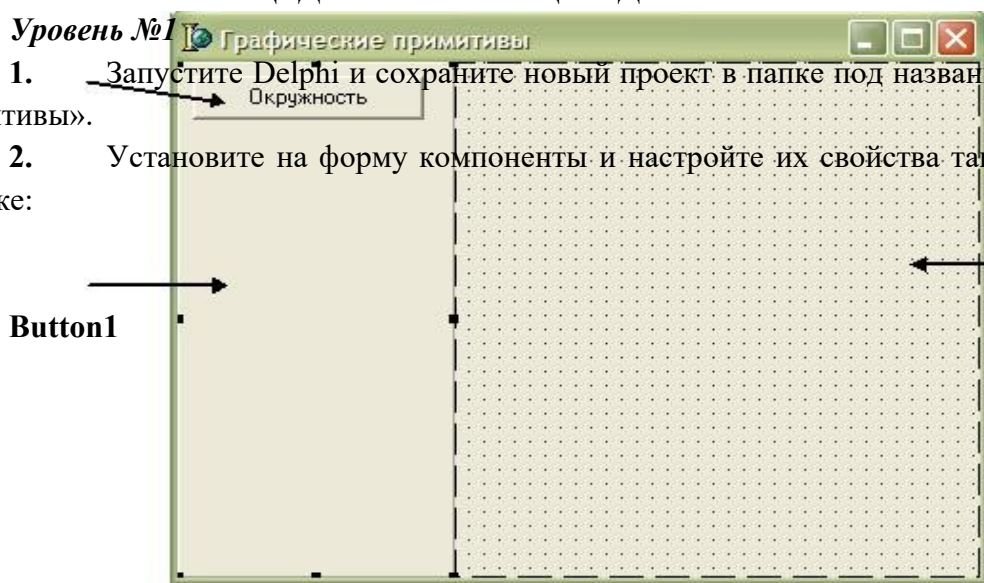
Уровень №3. Пользователь вводит 10 любых чисел в массив. Отсортировать этот массив по возрастанию и определить максимальный и минимальный элемент. Вывести исходный массив, отсортированный и найденные элементы.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №17.

ПРОЦЕДУРЫ И ФУНКЦИИ ДЛЯ РАБОТЫ С ГРАФИКОЙ

Уровень №1

1. Запустите Delphi и сохраните новый проект в папке под названием «Графические примитивы».
2. Установите на форму компоненты и настройте их свойства так, как показано на рисунке:



Panel1

PaintBox1

(на стр.

System)

3. Создайте обработчик события нажатия кнопки onClick и внесите в него следующий код программы:

```
with PaintBox1.Canvas do //ссылаемся на канву PaintBox1 begin  
Brush.Color:=clRed;//устанавливаем цвет заливки  
Brush.Style:=bsFDiagonal;//устанавливаем стиль заливки  
Pen.Color:=clBlue; //устанавливаем цвет границы  
Pen.Width:=4; //устанавливаем толщину границы  
Ellipse(10,10,100,100); //рисует эллипс  
end;
```

4. Установите на панель вторую кнопку, назовите ее «Задание 1». Эта кнопка будет также рисовать эллипс, но другого цвета (цвет заливки синий, сплошная, цвет границы черный). Создайте обработчик события нажатия кнопки и внесите следующий фрагмент программы, вставляя вместо многоточий свой код:

```
with PaintBox1.Canvas do //ссылаемся на канву PaintBox1 begin  
..... //цвет заливки - синий  
..... //стиль заливки - сплошная  
..... // цвет границы - черный  
.....//толщина границы – 3 пикселя Ellipse(10,10,100,100); //рисует эллипс  
end;
```

5. Установите на панель третью кнопку, назовите ее «Прямоугольник». Данная кнопка будет рисовать прямоугольник зеленого цвета. Внесите следующий код программы:

```
with PaintBox1.Canvas do //ссылаемся на канву PaintBox1 begin  
Brush.Color:=clGreen;//устанавливаем цвет заливки  
Brush.Style:=bsSolid;//устанавливаем стиль заливки  
Rectangle(10,10,100,100); //рисует прямоугольник end;
```

6. Запустите программу на выполнение (F9) и, щелкая по кнопкам, проверьте их работу.

7. Обратите внимание, если вы нажмете кнопку «Окружность», после нажатия на кнопку «Прямоугольник», окружность нарисована поверх прямоугольника. Чтобы устранить этот недостаток, добавьте вначале каждой процедуры следующие строки, которые также относятся к канве PaintBox1:

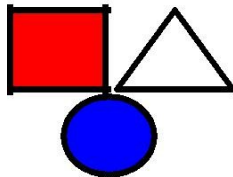
```
Brush.color:=clWhite;  
Rectangle(0,0,PaintBox1.Width,PaintBox1.Height);
```

8. Запустите программу и проверьте правильность работы кнопок.

9. Установите на форму четвертую кнопку и назовите ее «Задание 2». В обработчик события нажатия кнопки внесите следующий фрагмент, вместо точек подставляя нужный код:

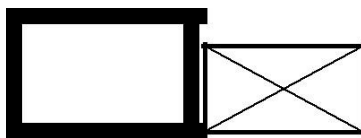
```
with PaintBox1.Canvas do //ссылаемся на канву PaintBox1 begin
//очищаем область рисования
Brush.color:=clWhite;
Rectangle(0,0,PaintBox1.Width,PaintBox1.Height);
//установите цвет заливки – синим, стиль заливки – сплошная
.....
Rectangle(10,10,100,100); //рисует прямоугольник
//установите толщину границы – 2, цвет заливки – красный
.....
Ellipse(100,10,200,100); //рисует эллипс
end;
```

10. Установите на панель еще одну кнопку и составьте программу, выводящую на экран следующее изображение:



Уровень №2

1. Напишите программу, выводящую рамку с конвертом. Рамка рисуется с помощью двух прямоугольников (оператор Rectangle) разного цвета, конверт – придумайте сами. Координаты тоже выберите сами.



2. Составьте программу, выводящую на экран елочку, состоящую из трех треугольников, поставленных друг на друга.

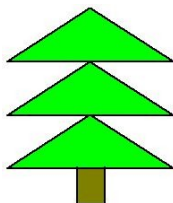
Для рисования треугольника используйте операторы MoveTo и LineTo.

Для заливки треугольника, используйте следующие операторы:

```
Brush.Color:=clgreen; //указывается цвет заливки - зеленый
FloodFill(100,30,clBlack,fsborder); //заливка до границы черного цвета, начиная с точки
```

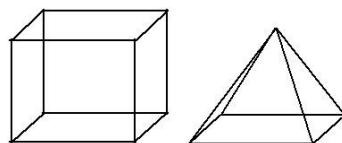
100,30

Для того, чтобы закрасить замкнутую область (например, треугольник), необходимо задать цвет заливки, а затем, используя оператор FloodFill, необходимо указать точку внутри этой области и цвет границы, до которой следует выполнить заливку.



Уровень №3

1. Создайте программу, выводящую на экран расположенные рядом куб и пирамиду.



2. Создайте программу, выводящую на экран веселую рожицу желтого цвета. Для рисования улыбки используйте процедуру Arc.

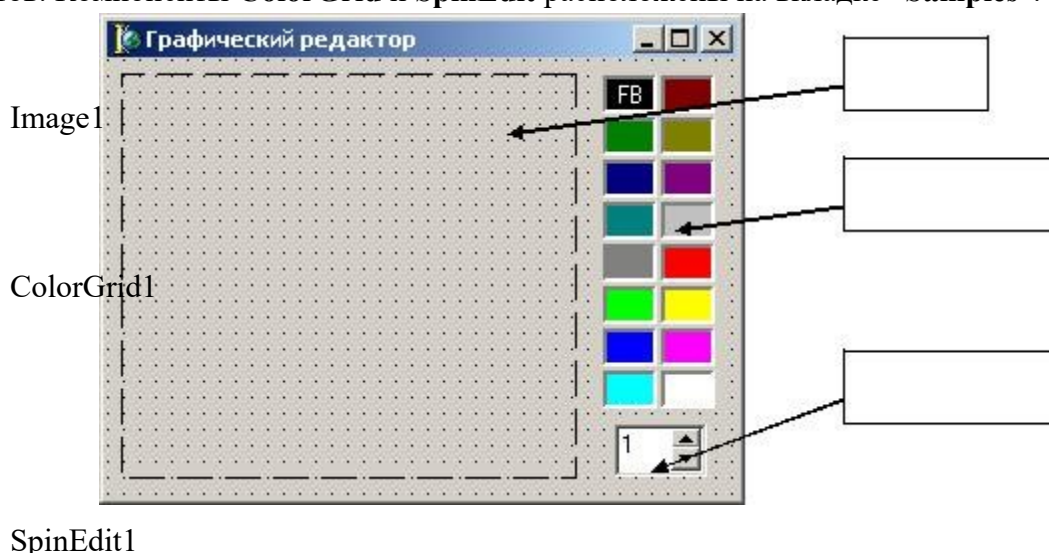


ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №18

СОЗДАНИЕ ГРАФИЧЕСКОГО РЕДАКТОРА

Начните новый проект в **Delphi** и сохраните его в отдельной папке под названием “**GrEdit**” (Имя модуля формы – **unit1.pas**, имя проекта – **GrEdit.dpr**).

Расположите на форме компоненты, как показано на рисунке и придайте им такой же внешний вид за счет изменения необходимых свойств в инспекторе объектов. Компоненты **ColorGrid** и **SpinEdit** расположены на вкладке “**Samples**”.



Создайте для компонента **Image1** обработчик события **OnMouseMove** и в созданный **Delphi** шаблон поместите следующую строку:

```
if ssLeft in Shift then Image1.Canvas.LineTo(X,Y);
```

Запустите проект и попробуйте перемещать мышью с нажатой левой кнопкой. За мышкой останется след. Отпустите кнопку — и рисование прекратится. Но стоит снова нажать левую кнопку мыши, как нарисованная кривая соединяется прямой линией с курсором мыши. Чтобы исправить ошибку: нужно в момент нажатия кнопки "перескакивать" в новое положение курсора без рисования.

Для этого создайте для компонента **Image1** обработчик события **OnMouseDown**:

```
Image1.Canvas.MoveTo(X, Y);
```

Обратите внимание, как работает компонент **ColorGrid** — при щелчке по нему левой кнопкой мыши устанавливается значок **FG** (ForegroundColor — цвет переднего плана, или цвет рисования), а при щелчке правой кнопкой мыши устанавливается значок **BG** (BackgroundColor — цвет фона).

ForegroundColor и **BackgroundColor** представляют собой свойства компонента **ColorGrid**, которые доступны программисту для чтения во время работы программы. Используем эти свойства для установки цвета рисования и цвета фона для компонента **Image1**.

При изменении компонента **ColorGrid1** будем изменять цвет пера и кисти для **Image1**. Для этого создайте для **ColorGrid1** обработчик события **onChange** и внесите в него следующие строки:

```
Image1.Canvas.Pen.Color:=ColorGrid1.ForegroundColor;  
Image1.Canvas.Brush.Color:= ColorGrid1.BackgroundColor;
```

А обработчик события **OnMouseDown** компонента **Image1** измените так:
with Image1.Canvas do

```
case Button of mbLeft : MoveTo(X,Y);  
mbRight: FloodFill (X,Y, Pixels [X,Y], fsSurface); end;
```

Для изменения толщины линии будем использовать компонент **SpinEdit1**. Установите у компонента **SpinEdit1** свойство **EditorEnabled** в **False** чтобы запретить редактирование с клавиатуры значения поля ввода. Создайте для **SpinEdit1** обработчик события **OnChange** и внесите следующий текст:

```
Image1.Canvas.Pen.Width:=SpinEdit1.Value;
```

Создайте для формы обработчик события **OnCreate** и внесите следующую строку:

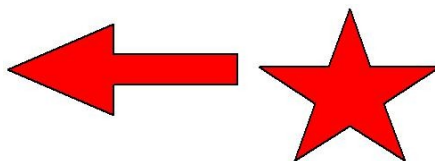
```
Image1.Canvas.FillRect(Image1.BoundsRect);
```

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №19

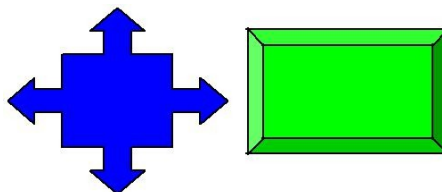
РИСОВАНИЕ ГЕОМЕТРИЧЕСКИХ ПРИМИТИВОВ

Уровень №1 .

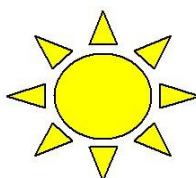
Составьте программу, выводящую на экран следующие элементы:



Уровень №2: Составьте программу, выводящую на экран следующие элементы:



Уровень №3: Составьте программу, выводящую на экран следующий элемент:



ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №20

ИСПОЛЬЗОВАНИЕ ЦИКЛИЧЕСКИХ АЛГОРИТМОВ И ПРОЦЕДУР ПРИ РАБОТЕ С ГРАФИКОЙ

Уровень №1

1. Запустите Delphi и сохраните новый проект в папке под названием «Практическая работа №3 - 1 уровень».
2. Установите на форму кнопку и компонент PaintBox (стр. System). Кнопке дайте название «Центрические окружности».
3. Создайте обработчик события нажатия кнопки и внесите в него следующий фрагмент программы:

```
var i:integer;  
with PaintBox1.Canvas do //указываем холст, на котором рисуем begin  
  for i:=15 downto 1 do //15 окружностей – цикл выполняется 15 раз begin  
    Brush.Color:=random(color); //цвет очередной окружности случайный  
    ellipse(100-5*i,100-5*i,100+5*i,100+5*i); //рисуем окружность  
  end;  
end;
```

4. Измените строчку программы
Brush.Color:=random(color);
на строчку
Brush.Color:=RGB(random(255),random(255),random(255));
5. Функция RGB получает цвет путем смешивания трех базовых цветов: Red, Green, Blue (красный, зеленый, синий). Каждый из цветов задается числом в диапазоне 0..255. Например, оператор Brush.Color:=RGB(255,0,0); задает красный насыщенный цвет, а строка Brush.Color:=RGB(255,0,255); задает фиолетовый цвет.

В нашем примере числа трех составляющих задаются случайным образом с помощью функции Random.

6. Запустите программу на выполнение.
7. Замените теперь, строку
Brush.Color:=RGB(random(255),random(255),random(255)); на строку
Brush.Color:=RGB(random(255),0,0);
8. Запустите программу и посмотрите изменения.
9. **Задание 1:** измените пример таким образом, чтобы на PaintBox выводились 15 окружностей зеленого оттенка.

10.Задание 2: измените пример так, чтобы вместо окружностей выводились 15 прямоугольников синего оттенка.

11.Задание 3: измените пример так, чтобы выводились не 15, а 20 прямоугольников.

Уровень №2

1. Создайте программу, выводящую на экран рисунок. 15 прямоугольников в ряд, цветовая гамма выбирается случайным образом. Для рисования использовать цикл.
2. Создайте программу, выводящую на экран шахматную доску. Использовать вложенный

цикл, который будет рисовать 8 строк состоящих из 8 квадратов.

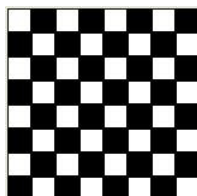
For i:=1 to 8 do

For j:=1 to 8 do

Begin

//выбирается цвет квадрата в зависимости от i и j

/ рисуется прямоугольник, координата X которого зависит от j, координата y зависит от i. End;



Уровень №3.

Создайте программу, которая рисует график функции $y=x^2+x$.

Предусмотреть для пользователя возможность настройки масштаба графика, границ.

Расчертить оси координат и создать подписи к ним координат).

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №21 МУЛЬТИМЕДИА. АНИМАЦИЯ В DELPHI

Уровень №1

1. Запустите Delphi и установите на форму кнопку и компонент PaintBox. В обработчике события нажатия кнопки внесите следующий код программы:

```
const r=15;  
var x:real;y0:integer; begin  
y0:=100;  
x:=0;  
with PaintBox1.Canvas do begin  
while x<PaintBox1.Width do begin brush.Color:=clred;  
Ellipse(round(x-r), y0-r, round(x+r),y0+r); Pen.Color:=clBtnFace;  
Ellipse(round(x-r), y0-r, round(x+r),y0+r); x:=x+0.1;  
end;  
end;  
end;
```

2. Запустите программу на выполнения.
3. **Задание 1:** измените программу таким образом, чтобы окружность двигалась с меньшей скоростью (медленней).
4. **Задание 2:** измените программу так, чтобы окружность двигалась по диагонали (из левого верхнего в правый нижний угол).
5. **Задание 3.** измените программу так, чтобы из левого верхнего угла в правый нижний угол двигался прямоугольник, меняя цвет случайным образом, а из правого верхнего угла в левый нижний двигался другой прямоугольник красного цвета.

Уровень №2.

Составьте программу, выводящую на экран, движущийся машину (или движущееся солнце).

Уровень №3.

Создайте короткий мультфильм на свободную тему.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №22 ФУНКЦИИ. РЕШЕНИЕ ЗАДАЧ.

Уровень №1.

Написать функцию, которая преобразует значение, заданное в градусах в значение, выраженное в радианах. ($n = \frac{\pi}{180} \cdot n \text{ рад}$) **Тест:** $n=5$, результат 0,0872 рад

Уровень №2

1. Разработать функцию, проверяющую, является ли число простым. С ее помощью найти и напечатать все простые числа из интервала от n до m .

Тест: $n=4$, $m=15$, результат: 5, 7, 11, 13

2. Написать функцию, которая для целого аргумента X возвращает количество цифр в его записи.

Тест: $x=1234$, результат: 4

Уровень №3.

Написать функцию, которая для целого аргумента X возвращает значение, полученное изменением порядка следования цифр на обратный.

Тест: $x=1234$, результат: $x=4321$

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №23 ПРОЦЕДУРЫ. РЕШЕНИЕ ЗАДАЧ

Уровень №1.

Создайте процедуру рисования объекта, закрашенного случайно выбранным цветом (рисунок 1). С помощью этой процедуры выведите на экран рисунок 2.



Рисунок 1

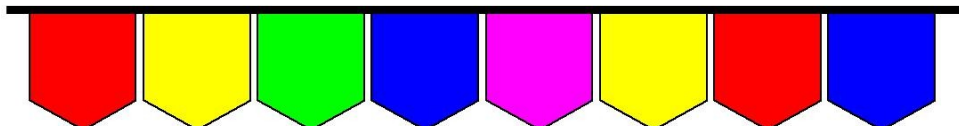


Рисунок 2

Уровень №2

1. Составить процедуру, заменяющую в исходной строке символов все единицы нулями и все нули единицами. Замена должна выполняться, начиная с заданной позиции строки.

Тест: s='111000101010', позиция n=4, Результат: '111111010101'

2. Составить процедуру, в результате обращения к которой из первой заданной строки удаляется каждый символ, принадлежащий и второй заданной строке.

Тест: s1='12345', s2='12678', результат: s1='345'.

Уровень №3. Составить процедуру, переводящую число из десятичной системы счисления в двоичную.

Тест: x=5, результат: x=101

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №24 МНОЖЕСТВЕННЫЙ ТИП ДАННЫХ. РЕШЕНИЕ ЗАДАЧ.

Уровень №1.

Задание №1: Составить программу, которая вырабатывает и выводит на экран дисплея наборы случайных чисел для игры в "Спортлото 5 из 36". Для заполнения каждой карточки спортлото необходимо получить набор из пяти псевдослучайных чисел. К этим числам предъявляются два требования:

1. числа должны находиться в диапазоне 1..36;
2. числа не должны повторяться.

Решение: Самостоятельно установите на форму компоненты, создайте обработчик события нажатия кнопки и внесите в него следующий код:

```
var
nb, k: Set of 1..36;
kol, l, i, n: Integer;
s:string;
begin
Randomize;
nb:=[1..36];
k:=[];
for l:=1 to 5 do
begin
repeat
n:=Random(36)
until (n in nb) and not (n in k);
k:=k+[n];
s:=s+', '+inttostr(n);
end;
memo1.Lines.Add(s);
end;
```

Задание №2: Заданы два слова. Определить буквы, которые не являются

общими для обоих слов.

Решение: Образует множества, содержащие буквы первого и второго слова. Затем найдем разности первого и второго, второго и первого множеств. Их объединение даст ответ.

```
procedure TForm1.Button4Click(Sender: TObject); type setchar = set of char;
const alf: string = 'абвгдеёжзийклмнопрстуфхцчщъыьэюя';//строка -
алфавит
var s1,s2:string; //заданные слова
i:integer;
ms1:setchar;
ms2:setchar;
g1,g2:setchar;
procedure print(mn:setchar);
var i:integer;
begin
for i:=1 to length(alf) do
if alf[i] in mn then memo1.Lines.Add(alf[i]);
end;
begin
s1:=edit2.Text;
s2:=edit3.text;
ms1:=[];
ms2:=[];
for i:=1 to length(s1) do ms1:=ms1+[s1[i]];
for i:=1 to length(s2) do ms2:=ms2+[s2[i]];
g1:=ms1-ms2;
g2:=ms2-ms1;
print(g1+g2);
end;
```

Уровень №2

1. Дана непустая последовательность символов. Построить и напечатать множества, элементами которого являются встречающиеся в последовательности цифры от '0' до '9' и знаки арифметических операций.

Тест: s='15g/8d0-630+g86', результат: множество = (15/80-63+)

2. Написать программу, которая печатает в убывающем порядке все цифры, входящие в запись заданного числа.

Тест: x=26143, результат 12346

Уровень №3.

Составить программу печати в алфавитном порядке всех букв текста, входящих в него более 2-х раз.

Тест: s='орлищиваеавцалли', результат 'аил'

Уровень №1.

Создать массив автовладельцев. Для каждого автовладельца известны номер, марка автомобиля, фамилия и адрес. Нужно подсчитать количество владельцев автомобилей заданной марки и вывести все сведения на экран.

Решение: Сведения об автовладельцах представим массивом записей. Исходные данные вводятся с клавиатуры. Работа с массивом записей аналогична работе с одномерным массивом.

Самостоятельно установите на форму необходимые компоненты и создайте обработчик для кнопки, в который внесите следующий код:

```
const nn=100;
Type mash = record
  nomer:integer;
  marka : string[20];
  fio:string[40];
  adres:string[60];
end;
var v:array[1..nn] of mash; k, n, i:integer; s:string; begin
  n:=StrToInt(Edit1.Text);
  for i:=1 to n do
    begin
      v[i].nomer:=i;
      v[i].fio:=InputBox('ФИО автовладельца','Введите ФИО','Петров');
      v[i].marka:=InputBox('Марка автомобиля','Введите марку','Mersedes');
      v[i].adres:=InputBox('Адрес автовладельца','Введите адрес','Калинина 35-52'); end;
      s:=InputBox('Введите марку','Какая марка вас интересует?', 'Mersedes');
      k:=0;
      for i:=1 to n do
        if v[i].marka = s then
          begin
            with v[i] do Memo1.Lines.Add(inttostr(nomer)+' '+marka+' - '+adres); k:=k+1;
          end;
      Memo1.Lines.Add('Результат: '+inttostr(k));
    end;
```

Уровень №2. Известны ФИО абонента, дата выдачи книги, автор, издательство, название, год издания и цена книги. Составить программу, позволяющую внести данные и выяснить следующее:

- a. Сколько абонентов пользовалось данной книгой;
- b. Сколько абонентов поставлено на учет данного числа;
- c. Получить список абонентов, на руках у которых книга стоимости не ниже заданной.

Уровень №3. Создан массив автовладельцев. Для каждого автовладельца известны номер, марка автомобиля, фамилия и адрес. Написать программу, выводющую фамилии и адреса автовладельцев, имеющих более одного автомобиля.

Уровень №1.

Известны ФИО сотрудника, его адрес (улица, номер дома, номер квартиры), телефон, пол и возраст. Составить программу, позволяющую внести данные и определить:

- a. сколько сотрудников мужского и сколько сотрудников женского пола проживают по заданной улице.
- b. сколько детей, в возрасте до 18 лет проживают в заданном доме;

Уровень №2

1. Создан массив автовладельцев. Для каждого автовладельца известны номер, марка автомобиля, фамилия и адрес. Написать программу, выясняющую, сколько машин разных марок имеется.
2. Известны номер рейса, ФИО пассажира, багаж (количество вещей), вес багажа (в кг.). Определить:
 - a. пассажира (-ов), багаж которого состоит из одной вещи весом меньше 20 кг,
 - b. средний вес багажа,
 - c. количество пассажиров, вес багажа которых превосходит средний.

Уровень №3. Известны ФИО студента и оценки по четырем предметам (математика, информатика, программирование, философия). Определить:

- a. сколько студентов имеют неудовлетворительную оценку хотя бы по одному предмету;
- b. сколько студентов сдали все экзамены на «5»;
- c. средний балл группы по философии;
- d. какое количество студентов имеет средний балл от 4 до 5;
- e. какой процент студентов не имеет задолженностей;

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №27 ПРОЦЕДУРЫ ДЛЯ РАБОТЫ С ФАЙЛАМИ

Уровень №1.

Составить программу, которая создает текстовый файл и выводит в него несколько строк, а также позволяет считать эти данные в компонент Мемо.

Решение: Установите на форму компонент Мемо и две кнопки: «Записать» и «Считать».

В обработчик кнопки «Записать» внесите следующий код программы:

```
Var f:textFile; i:integer;  
Begin  
Assignfile(f, ExpandFileName('1.txt'));  
Rewrite(f);  
For i:=1 to 3 do Writeln(f, 'строка '+inttostr(i));  
CloseFile(f);  
End;
```

В обработчик кнопки «Считать» внесите следующий код.

```
Var f:textFile; i:integer;
```

```

Begin
Assignfile(f, ExpandFileName('1.txt'));
Reset(f);
While not EOF do
begin
Readln(f,s);
Memo1.lines.add(s);
End;
CloseFile(f);
End;

```

Уровень №2

1. Написать программу, которая выводит текст из файла на экран, при этом подставляет номера строк в начало каждой строки.
2. Написать программу, которая считывает исходный текст этой программы и подсчитывает, сколько раз в этой программе встречаются слова BEGIN и END.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №28 ФУНКЦИИ ДЛЯ РАБОТЫ С ФАЙЛАМИ

Уровень №3.

Написать программу, которая при первом запуске создает текстовый файл, а при последующих – дописывает в конец этого файла информацию, содержащую: фамилию, имя, дату рождения, домашний адрес. Для определения существования файла используйте функцию *FileExists*:

```

if FileExists('C:\Мои документы\1.txt') then //существует файл else // не
существует

```

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №29 ТЕКСТОВЫЕ ФАЙЛЫ. РЕШЕНИЕ ЗАДАЧ

Уровень №1.

Написать программу, которая считывает исходный текст этой программы и определяет, содержатся ли в нем буквы английского алфавита, и подсчитывает количество каждой буквы.

Уровень №2.

Написать программу, которая считывает исходный текст этой программы и записывает в другой текстовый файл только те слова, в которых есть буква Е(все равно Большая или маленькая).

Уровень №3. Чтобы зашифровать текст, записанный с помощью русских букв и знаков препинания, его можно переписать, заменив каждую букву непосредственно следующей за ней по алфавиту (буква «я» заменяется на «а»). Составить программу, шифрующую текстовый файл и выводящую его на экран.

Основная литература:

1. Зыков С. В. Введение в теорию программирования. Объектно-ориентированный подход М.: Национальный Открытый Университет «ИНТУИТ», 2016 Объем: 189 Дополнительная информация: 2-е изд., испр.
2. Мейер Б. Объектно-ориентированное программирование и программная инженерия М.: Национальный Открытый Университет «ИНТУИТ», 2016 Объем: 286 Дополнительная информация: 2-е изд., испр.

Дополнительная литература:

1. Кулямин В. Компонентный подход в программировании М.: Национальный Открытый Университет «ИНТУИТ», 2016 Объем: 591 Дополнительная информация: 2-е изд., исправ.