



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**
(ДГТУ)



УТВЕРЖДАЮ

Проректор по УриНО

С.В. Пономарева

«*02*» 2023 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по факультативу**

Ф.01 Основы программирования

образовательной программы

по специальности среднего профессионального образования

33.02.01 Фармация

Ростов-на-Дону
2023 г.


Лист согласования

Фонд оценочных средств по факультативу разработан на основе Федерального государственного образовательного стандарта (далее – ФГОС) по специальности среднего профессионального образования (далее - СПО) 33.02.01 Фармация

Разработчик(и):

Преподаватель

Авиационно-технологического колледжа


_____ А.А. Юрина

Фонд оценочных средств рассмотрен и одобрен на заседании цикловой комиссии Авиационно-технологического колледжа, протокол № 5 от 06.02.2023 г.

Председатель цикловой комиссии _____ А.Ю. Коротенко

« 06 » 02 2023 г.**Согласовано:****Рецензенты:**

ООО «Ромашка»,
г. Ростов-на-Дону
(место работы)

заведующий
аптекой
(занимаемая должность)

В.И.Заритовская
(инициалы, фамилия)


Авиационно-технологический
колледж ДГТУ
(место работы)

преподаватель высшей
квалификационной категории
(занимаемая должность)

Н.С. Осмоловская
(инициалы, фамилия)

Одобен на заседании педагогического совета Авиационно-технологического колледжа, протокол № 3 от 09.02.2023 г.

Председатель педагогического совета


_____ В.А. Зибров

Содержание

1	Паспорт Фонда оценочных средств	4
1.1	Область применения Фонда оценочных средств	4
1.2	Требования к результатам освоения факультатива	4
2	Результаты освоения факультатива, подлежащие проверке	6
2.1	Показатели оценки результатов обучения	6
3	Фонд оценочных средств	8
3.1	Текущий контроль успеваемости	8
3.2	Промежуточная аттестация	15

1 Паспорт Фонда оценочных средств

1.1 Область применения Фонда оценочных средств

Фонд оценочных средств разработан в соответствии с требованиями ФГОС специальности СПО 33.02.01 Фармация и рабочей программой факультатива Ф.01 Основы программирования.

Фонд оценочных средств предназначен для аттестации обучающихся на соответствие их персональных достижений поэтапным требованиям образовательной программы (ОП) по специальности среднего профессионального образования 33.02.01 Фармация. ФОС включает в себя контрольно-оценочные материалы, которые позволяют оценить общие и профессиональные компетенции, умения, знания, личностные результаты.

Факультатива, в соответствии с учебным планом, изучается на первом курсе во втором семестре и завершается другими формами (тестирование).

1.2 Требования к результатам освоения факультатива

Перечень компетенций, формируемых в процессе изучения факультатива.

Рабочей программой факультатива Ф.01 Основы программирования предусмотрено формирование следующих компетенций:

ОК 1: Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 2: Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 4: Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 9: Использовать информационные технологии в профессиональной деятельности.

ПК 1.4: Осуществлять розничную торговлю и отпуск лекарственных препаратов населению, в том числе по льготным рецептам и требованиям медицинских организаций.

ПК 1.6: Осуществлять оптовую торговлю лекарственными средствами и другими товарами аптечного ассортимента.

ПК 1.7: Оформлять первичную учетно-отчетную документацию.

В результате освоения факультатива обучающиеся должны знать:

31 - основные понятия автоматизированной обработки информации;

32 - общий состав и структуру персональных компьютеров;

33 - состав, функции и возможности использования информационных и телекоммуникационных технологий в профессиональной деятельности;

34 - методы и средства сбора, обработки, хранения, передачи и накопления информации

35 - базовые системные программные продукты и пакеты прикладных программ в области профессиональной деятельности;

36 - основные методы и приемы обеспечения информационной безопасности;

37 - понятие системы программирования

В результате освоения факультатива обучающиеся должны уметь:

У1 - использовать технологии сбора, размещения, хранения, накопления, преобразования и передачи данных в профессионально ориентированных информационных системах;

У2 - использовать в профессиональной деятельности различные виды программного обеспечения, в том числе специального;

У3 - применять компьютерные и телекоммуникационные средства;

У4 - использовать языки программирования, строить логически правильные и эффективные программы.

В соответствии с рабочей программой воспитательной работы и календарным планом воспитательной работы в процессе изучения факультатива формируются следующие личностные результаты:

ЛР 4: Проявляющий и демонстрирующий уважение к людям труда, осознающий ценность собственного труда. Стремящийся к формированию в сетевой среде лично и профессионального конструктивного «цифрового следа»;

ЛР 10: Заботящийся о защите окружающей среды, собственной и чужой безопасности, в том числе цифровой;

ЛР 20: Владеющий навыками адаптации в динамично изменяющемся и развивающемся мире.

2 Результаты освоения факультатива, подлежащие проверке

2.1 Показатели оценки результатов обучения

Основные показатели и критерии оценки результатов освоения факультатива представлены в таблице.

Результаты освоения (объекты оценивания)	Основные показатели оценки результата	Критерии оценки результата	Тип задания;	Форма аттестации (в соответствии с учебным планом)
ОК 1, 31, 32, 33, 34, ЛР 10, ЛР20	Способность оценивать эффективность и качество выполнения профессиональных задач, формулировать цели и задачи профессиональной деятельности. Воспроизведение целей и методов при решении профессиональных задач; способов самоорганизации, процесса постановки цели, выбора и применения методов при решении профессиональных задач.	Оценка умения планировать, организовывать и оценивать собственную деятельность, понимать суть профессиональных задач, оценивать и анализировать процесс и результат.	устные опросы; письменные опросы; решение тестовых заданий, выполнение практических работ	Тестирование
ОК 2, 35, 36, У1, ЛР4, ЛР20	Использование различных источников, включая электронные ресурсы, медиа-ресурсы, Интернет-ресурсы, периодические издания по специальности для решения профессиональных задач.	Оценка умения: - находить и использовать разнообразные источники информации; - грамотно определять тип и форму необходимой информации; - получать нужную информацию и сохранять ее в		

		удобном для работы формате; - определять степень достоверности и актуальности информации; - упрощать подачу информации для ясности понимания и представления.		
ОК 4, ЛР4	Понимание основ работы в коллективе и в команде. Понимание общих целей. Координация своих действий с другими участниками общения; работа в коллективе и команде, эффективное общение с коллегами, руководством.	Экспертное наблюдение и оценка результатов формирования поведенческих навыков в ходе обучения. Демонстрация готовности и способности вести диалог с другими людьми. Оценка навыков владения профессиональной лексикой, соблюдения этических норм поведения.		
ОК 09, ПК1.4 ПК 1.6 35, 36, 37, У2, У3, У4, ЛР 20	Демонстрация навыков использования программного обеспечения для осуществления оптовой и розничной торговли лекарственными средствами и другими товарами аптечного ассортимента.	Оценка правильности работы с программным обеспечением для осуществления оптовой и розничной торговли лекарственными средствами и другими товарами аптечного ассортимента.		

ОК 09, ПК 1.7, 35, 36, 37, У2, У3, У4, ЛР 20	Понимание языков программирования для работы с первично-учетно-отчетной документацией	Оценка демонстрации навыков программирования для работы с первично-учетно-отчетной документацией		
--	---	--	--	--

3 Фонд оценочных средств

3.1 Текущий контроль успеваемости

Текущий контроль проводится с целью установления соответствия достижений, обучающихся требованиям образовательной программы к результатам обучения и формирования компетенций, обеспечения своевременной обратной связи, для коррекции обучения, активизации самостоятельной работы обучающихся. Текущий контроль успеваемости осуществляется в ходе повседневной учебной работы по факультативу.

Перечень оценочных средств

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Тест	Форма контроля, направленная на проверку уровня освоения контролируемого теоретического материала по дидактическим единицам факультатива (терминологический аппарат, основные методы, информационные технологии, приемы, документы)	Тестовые задания по темам факультатива
2	Письменный опрос	Средство для проверки умений применять полученные знания по освоенной теме факультатива. Рекомендуется для оценки знаний, умений и владений обучающихся	Контрольные задания по темам факультатива
3	Вопросы для обсуждения на занятиях (устный опрос)	Вопросы для обсуждения, необходимые для контроля усвоения теоретических знаний. Используется при проведении фронтального опроса по темам факультатива.	Перечень вопросов для обсуждения по темам факультатива
4	Практическая работа	Средство проверки знаний, умений и приобретения практического опыта. Задание, в котором обучающемуся предлагается решить практические производственной задачи. Рекомендуется для оценки знаний, умений и практического опыта.	Комплекты практических заданий

3.1.1 Тестирование (Демонстрационный вариант)

Раздел 1. Основы алгоритмизации

Задания нацелены на проверку знания и понимания основных понятий по данному разделу
Форма контроля - тестирование.

Задание. Выполните тестовое задание

Общие рекомендации по выполнению теста

Внимательно прочитайте задание, выберите правильные варианты ответа.

1) Алгоритм – это...

- А) протокол вычислительной сети;
- Б) правила выполнения определенных действий;
- В) описание последовательности действий, строгое исполнение которых приводит к решению поставленной задачи за конечное число шагов;
- Г) ориентированный граф, указывающий порядок выполнения некоторого набора команд;
- Д) набор команд для компьютера.

2) Алгоритм включает в себя ветвление, если.

- А) он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий;
- Б) он представим в табличной форме;
- В) его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий;
- Г) ход его выполнения зависит от истинности тех или иных условий;
- Д) он включает в себя вспомогательный алгоритм.

3) Свойство алгоритма, заключающиеся в отсутствие ошибок (алгоритм должен приводить к правильному результату для всех допустимых входных значениях), называется.

- А) Массовость;
- Б) Конечность;
- В) Дискретность;
- Г) Результативность;
- Д) Детерминированность.

4) Свойство алгоритма, заключающиеся в том, что один и тот же алгоритм можно использовать с разными исходными данными, называется.

- А) детерминированность
- Б) массовость
- В) конечность
- Г) дискретность
- Д) результативность

5) Какая структура программы верная. Перечислите и объясните найденные ошибки.

А) Begin
program MyFirst;
X:=Y+195;
end.

Б) program MyProg;
begin
Writeln ('Привет');
end.

6) Какое значение будет принимать переменная X, после выполнения фрагмента программы. Запишите решение и ответ: f:=17;

d:=5;

If f>=d then x:=f else x:=d

A) 5;

Б) 12;

В) 2;

Г) 17.

7) Чему будет равно значение переменной С в результате выполнения серии операторов.

Запишите решение и ответ.

a := 6*12 + 3;

b := a div 10 + 5;

a := b mod 10 + 1;

c := a*a + b*b – a / 2 * b.

8) Найдите ошибки в записях оператора.

A) Write ©;

Б) Writeln;

В) Writeln (Введите любое число);

Г) Write X,Y;

Д) WRITELN ('воскресенье, нерабочий день').

9) Определите, что будет напечатано в результате работы следующего фрагмента программы

Запишите решение и ответ.

```
var k, s: integer;
```

```
begin
```

```
s:=0;
```

```
k:=1;
```

```
while k < 11 do begin
```

```
s:=s+k;
```

```
k:=k+1;
```

```
end;
```

```
write(s);
```

```
end.
```

10) Составьте блок-схему, напишите программу на языке Паскаль.

Задача: Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае вычесть из него 2. Вывести полученное число.

Критерии оценивания (за правильный ответ дается 1 балл)

от 0 % до 70 % включительно – оценка «**неудовлетворительно**»;

от 71% до 80% включительно – оценка «**удовлетворительно**»;

от 81 % до 90 % включительно – оценка «**хорошо**»;

от 91 % до 100 % включительно – оценка «**отлично**»

3.1.2 Письменный опрос

Тема 3.1 Понятие модуля. Описание модулей. Принципы модульного программирования. Построение программ на основе модулей. Пользовательский модуль.

Задания нацелены на проверку знания и понимания основных понятий по данной теме

Форма контроля – письменный опрос.

1. Программный модуль.

2. Прочность модуля.

3. Технологичность программного обеспечения.

4. Оптимизация программ.
5. Функциональная декомпозиция разбиения.

Критерии оценивания:

Оценка 5 «отлично»	обучающийся показывает глубокие осознанные знания по освещаемому вопросу, владение основными понятиями, терминологией; владеет конкретными знаниями, умениями по данному факультативу; ответ полный доказательный, четкий, грамотный, иллюстрирован практическим опытом профессиональной деятельности
Оценка 4 «хорошо»	обучающийся показывает глубокое и полное усвоение содержания материала, умение правильно и доказательно излагать программный материал, допускает отдельные незначительные неточности в форме и стиле ответа.
Оценка 3 «удовлетворительно»	обучающийся понимает основное содержание учебной программы, умеет показывать практическое применение полученных знаний. Вместе с тем допускает отдельные ошибки, неточности в содержании и оформлении ответа; ответ недостаточно последователен, доказателен и грамотен.
Оценка 2 «неудовлетворительно»	обучающийся имеет существенные пробелы в знаниях, допускает ошибки, неточности в содержании рассказываемого материала, не выделяет главного, существенного в ответе. Ответ поверхностный, бездоказательный, допускаются речевые ошибки.

3.1.3 Вопросы для обсуждения на занятиях (устный опрос)

Задания нацелены на проверку знания и понимания основных понятий по данным разделам

Тема 1.2 Методы разработки и способы представления алгоритмов.

1. Виды алгоритмов
2. Способы представления алгоритмов.
3. Основные принципы методов ветвей и границ.
4. Программная форма записи алгоритмов.
5. Дерево решений.

Критерии оценивания:

Оценка 5 «отлично»	обучающийся показывает глубокие осознанные знания по освещаемому вопросу, владение основными понятиями, терминологией; владеет конкретными знаниями, умениями по данному факультативу; ответ полный доказательный, четкий, грамотный, иллюстрирован практическим опытом профессиональной деятельности
Оценка 4 «хорошо»	обучающийся показывает глубокое и полное усвоение содержания материала, умение правильно и доказательно излагать программный материал, допускает отдельные незначительные неточности в форме и стиле ответа.
Оценка 3 «удовлетворительно»	обучающийся понимает основное содержание учебной программы, умеет показывать практическое применение полученных знаний. Вместе с тем допускает отдельные ошибки, неточности в содержании и оформлении ответа; ответ недостаточно последователен, доказателен и грамотен.
Оценка 2 «неудовлетворительно»	обучающийся имеет существенные пробелы в знаниях, допускает ошибки, неточности в содержании рассказываемого материала, не выделяет главного, существенного в ответе. Ответ поверхностный, бездоказательный, допускаются речевые ошибки.

3.1.5 Практическое задание (Демонстрационный вариант)

Тема 2.7. Создание записей и массивов записей

Оснащение: OS Windows, MS Office.

Формируемые компетенции: ОК 01 ОК 09 ПК 1.7 ЛР 20

Теоретическая часть

Скалярный тип – простой тип данных. Скалярное данное неделимо. Массив – это структурированный тип данных. Массив состоит из нескольких элементов. Ко всему массиву можно обращаться по его имени. Можно обращаться к его элементу, но для этого надо задать индекс (индексы). Массивы бывают одномерные и многомерные. Для объявления массива необходимо задать типы его индексов и компонент:

```
ARRAY [Тип индексов] OF <Тип компонент>;
```

Тип компонент массива – это просто тип данных, ассоциированный с каждой компонентой массива. Тип компонент может быть любым REAL, INTEGER, CHAR, BOOLEAN, перечислимым, интервальным. В качестве компоненты массива может быть взят и тип массив.

Тип индекса должен быть одним из упорядоченных типов, т.е. любым скалярным типом, кроме REAL: INTEGER, CHAR, интервальный, перечислимый. Тип индекса определяет границы изменения индекса. Если сделана попытка использовать несуществующую компоненту, то возникает ошибка (ошибка неверного индекса).

Одномерные массивы.

Одномерный массив можно задать (объявить) двумя способами:

1. С помощью служебного слова TYPE описывается тип массива, а затем с помощью VAR вводится переменная этого типа.

Общая форма записи

```
TYPE <тип массива> = ARRAY [тип индекса] OF <тип компонент>;
```

```
VAR <переменная>: <тип массива>;
```

2. С помощью слова VAR сразу описывается переменная типа массив.

Общая форма записи

```
VAR <переменная>: ARRAY [тип индекса] OF <тип компонент>;
```

Например, объявление массива из 100 элементов типа REAL можно осуществить двумя способами:

```
1. type R100 = array [1..100] of real;
```

```
var A: R100;
```

```
2. var A: array [1..100] of real.
```

Здесь задан массив с именем A, и его элементы имеют имена: A[1],...,A[100]. Чаще всего для типа индекса используют интервальный тип на основе типов INTEGER и CHAR. Однако можно в качестве индексов брать перечислимый тип.

Пример 1. Подсчет числа вхождений букв в текст определенной длины.

```
program COUNTER;
```

```
var COUNT: array ['a'..'z'] of integer;
```

```
CH: char; N: integer;
```

```
begin
```

```
for CH := 'a' to 'z' do
```

```
  COUNT [CH] := 0; N := 0;
```

```
repeat
```

```
  read (CH); N := N + 1;
```

```
  if (CH >= 'a') and (CH <= 'z') then
```

```
    COUNT [CH] := COUNT [CH] + 1;
```

```
until CH = '!';
```

```

for CH := 'a' to 'z' do
writeln (CH, COUNT [CH]:5);
end.

```

Пояснение. В этом примере тип индекса есть интервальный тип на базе типа CHAR, а тип компонент есть целое число. Таким образом, элементы массива – числа, а их индексы – буквы, т.е. число элементов массива равно 26 (по числу букв латинского алфавита).

Рассмотрим теперь случай, когда тип индекса задан перечислимым типом, а компоненты массива представлены компонентами интервального типа на базе типа INTEGER.

Пример 2. Присваивание переменной с именем месяца числа дней этого месяца.

```

program NUMBRDAY;
type MONAT = (JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG,
SEP, OKT, NOV, DEC);
var DAY : array [MONAT] of 28..31; T : MONAT;
begin
for T := JAN to DEC do
case T of
JAN, MAR, MAY, JUL, AUG, OKT, DEC: DAY [T] := 31;
APR, JUN, SEP, NOV: DAY [T] := 30;
FEB : DAY [T] := 28;
end;
end.

```

Практическая часть

Дано описание массива: `constb:array[1..2]ofinteger=(2,11,8,64,101)`, Определить способ заполнения массива.

fori:=1to13doread(a[i]);
с помощью оператора присваивания (случайными числами)
ввод каждого значения с клавиатуры
с помощью оператора присваивания (по формуле)
На рисунке 1 показаны способы ввода массива.

45 21 56 43 83 56 69 34 28 15

Рисунок 1 – Способы ввода массива

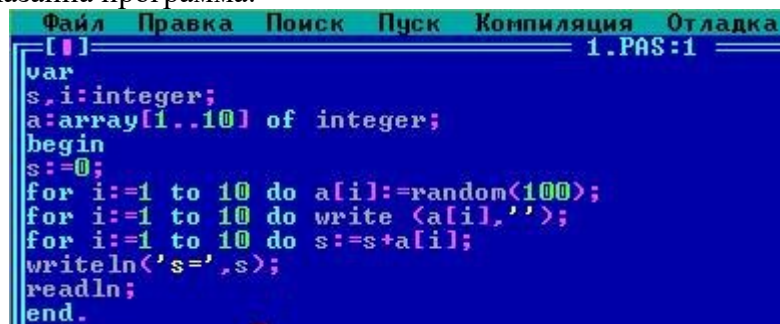
Элементы массива выводятся в строку вывод массива с комментариями

Выбери верный ответ.

Под _____ (упорядочением) массива понимают перераспределение значений его элементов в некотором определенном порядке.

- А) помощь;
- Б) обыкновенный;
- В) сортировка;
- Г) запуск программы.

На рисунке 2 показанна программа.



```

Файл  Правка  Поиск  Пуск  Компиляция  Отладка
[ ] 1.PAS:1
var
s,i:integer;
a:array[1..10] of integer;
begin
s:=0;
for i:=1 to 10 do a[i]:=random(100);
for i:=1 to 10 do write (a[i],',');
for i:=1 to 10 do s:=s+a[i];
writeln('s=',s);
readln;
end.

```

Рисунок 2 – Программа

Запиши данную программу на языке программирования Паскаль. Запусти программу и запиши верный ответ.

Ответ: s=

Задание 1. В двумерном массиве, состоящем из целочисленных элементов, в каждом столбце поменять местами наибольший по модулю и последний не принадлежащий интервалу (a, b) элементы массива, найти наибольший по модулю элемент столбца.

Задание 2. Составить программу, позволяющую в одномерном массиве, состоящем из N вещественных элементов, вычислить:

- А) сумму модулей отрицательных элементов массива;
- Б) количество элементов массива, не принадлежащих интервалу (a, b);
- В) наименьший из элементов массива, принадлежащих отрезку [a, b];
- Г) количество элементов массива, равных первому элементу.

Задание 3. двумерном массиве, состоящем из целочисленных элементов, поменять местами:

- А) в каждом столбце наибольший по модулю и последний положительный элементы;
- Б) в каждом столбце первый и последний отрицательные элементы;
- В) в каждой строке наибольший и наименьший элементы;
- Г) в каждом столбце первый принадлежащий отрезку [a, b] и первый отрицательный элементы;

Задания для самостоятельной проверки:

1. Как описываются в языке Паскаль одномерный и двумерные массивы?
2. Может ли массив содержать разнотипные данные?
3. Привести пример массива, описание которого выглядит следующим образом: `var A: array [1..3, 20..24] of real..`
4. Можно ли при обработке двумерных массивов использовать однократные циклы? Если да, то приведите примеры.
5. Если в одномерном массиве проверяется «похожесть» его первой и второй части, то в каких границах надо писать оператор `for` для «прохождения» по этому массиву?

Критерии оценивания при текущем контроле освоения практических навыков и умений:

Оценка «**отлично**» – обучающийся обладает системными теоретическими знаниями (знает методику выполнения практических навыков, показания и противопоказания, возможные осложнения, нормативы и проч.), без ошибок самостоятельно демонстрирует выполнение практических умений,

Оценка «**хорошо**»— обучающийся обладает теоретическими знаниями (знает методику выполнения практических навыков, показания и противопоказания, возможные осложнения, нормативы и проч.), самостоятельно демонстрирует выполнение практических умений, допуская некоторые неточности (малосущественные ошибки), которые самостоятельно обнаруживает и быстро исправляет,

Оценка «**удовлетворительно**» — обучающийся обладает удовлетворительными теоретическими знаниями (знает основные положения методики выполнения практических навыков, показания и противопоказания, возможные осложнения, нормативы и проч.), демонстрирует выполнение практических умений, допуская некоторые ошибки, которые может исправить при коррекции их преподавателем,

Оценка «**неудовлетворительно**» — обучающийся не обладает достаточным уровнем теоретических знаний (не знает методики выполнения практических навыков, показаний и противопоказаний, возможных осложнений, нормативы и проч.) и/или не может самостоятельно продемонстрировать практические умения или выполняет их, допуская грубые ошибки.

3.2 Промежуточная аттестация

Учебным планом специальности 33.02.01 Фармация, форма контроля по факультативу «Основы программирования» - другие формы (тестирование).

Итоговое занятие проводится за счет времени, отведенного на изучение факультатива.

Контроль знаний проводится на последнем занятии по данному факультативу, одновременно для всей учебной группы.

В результате аттестации по факультативу осуществляется комплексная проверка умений и знаний, динамику формирования общих и профессиональных компетенций, а также личностных результатов.

Тест

1. Присоединение частицы НЕ к высказыванию – это...

- А) дизъюнкция;
- Б) конъюнкция;
- В) импликация;
- Г) эквивалентность;
- Д) инверсия.

2. Соединение двух простых высказываний А и В в одно составное с помощью союза И – это...

- А) дизъюнкция;
- Б) конъюнкция;
- В) импликация;
- Г) эквивалентность;
- Д) инверсия.

3. Операция дизъюнкция называется иначе

- А) логическое умножение;
- Б) логическое сложение;
- В) логическое следование;
- Г) логическое равенство;
- Д) логическое отрицание.

4. Операция импликация называется иначе.

- А) логическое умножение;
- Б) логическое сложение;
- В) логическое следование;
- Г) логическое равенство;
- Д) логическое отрицание.

5. Эквивалентность – это:

- А) соединение двух простых высказываний в одно составное с помощью союза И;
- Б) соединение двух простых высказываний в одно составное с помощью союза ИЛИ;
- В) соединение двух высказываний в одно с помощью оборота речи «Если ..., то...»;
- Г) соединение двух высказываний в одно с помощью оборота речи «тогда и только тогда, когда.»;
- Д) присоединение частицы НЕ к высказыванию;

6. Составное высказывание, образованное с помощью операции импликации.

- А) ложно тогда и только тогда, когда из истинной предпосылки следует ложный вывод;
- Б) истинно тогда и только тогда, когда из истинной предпосылки следует ложный вывод;
- В) истинно тогда и только тогда, когда оба высказывания одновременно либо ложны, либо истинны;
- Г) истинно, когда хотя бы одно высказывание истинно;
- Д) истинно тогда и только тогда, когда составляющие высказывания одновременно истинны.

7. Составное высказывание, образованное с помощью операции эквивалентности.

- А) ложно тогда и только тогда, когда из истинной предпосылки следует ложный вывод;

- Б) истинно тогда и только тогда, когда из истинной предпосылки следует ложный вывод;
- В) истинно тогда и только тогда, когда оба высказывания одновременно либо ложны, либо истинны;
- Г) истинно, когда хотя бы одно высказывание истинно;
- Д) истинно тогда и только тогда, когда составляющие высказывания одновременно истинны.

8. Какой логической операции соответствует таблица истинности?

- А) дизъюнкция;
- Б) конъюнкция;
- В) эквивалентность;
- Г) инверсия;
- Д) импликация.

9. Служебное слово VAR в программе на языке Pascal фиксирует начало раздела программы, содержащего.

- А) операторы;
- Б) список меток;
- В) описание сложных типов данных;
- Г) перечень констант;
- Д) описание переменных.

10. В алфавит языка Pascal не входит служебное слово.

- А) THEN;
- Б) BEGIN;
- В) END;
- Г) STEP;
- Д) IF.

11. Числа в языке Pascal различаются.

- А) как натуральные и целые;
- Б) как целые и вещественные;
- В) как натуральные и вещественные;
- Г) как целые и иррациональные;
- Д) как целые и рациональные.

12. Формальные параметры процедуры.

- А) описываются в ее заголовке;
- Б) перечисляются при вызове процедуры;
- В) указываются при описании данных в программе;
- Г) декларируются при описании промежуточных переменных процедуры;
- Д) присваиваются в процессе выполнения программы.

13. В системе Turbo Pascal начало раздела программы, содержащего список меток, фиксируется служебным словом:

- А) CONST;
- Б) TYPE;
- В) VAR;
- Г) LABEL;
- Д) BEGIN.

14. Оператор организации ввода данных с клавиатуры в системе программирования Turbo Pascal записывается с использованием служебного слова.

- А) WRITE;
- Б) INPUT;
- В) READ;
- Г) DEFFN;
- Д) RESET.

15. Комментарий к тексту программы на языке Pascal заключается.

- А) в фигурные скобки;

- Б) в круглые скобки;
- В) в квадратные скобки;
- Г) в апострофы;
- Д) между служебными словами Begin, End.

16. Служебное слово CONST в программе на языке Pascal фиксирует начало раздела программы, содержащего.

- А) перечень констант;
- Б) список меток;
- В) описание переменных;
- Г) операторы;
- Д) описание сложных типов данных.

17. В качестве имени в языке Pascal нельзя использовать.

- А) BR;
- Б) OR;
- В) WR;
- Г) AR;
- Д) DR.

18. При записи оператора вывода на языке Pascal используется служебное слово.

- А) WRITE;
- Б) RESET;
- В) READ;
- Г) BEGIN;
- Д) WHILE.

19. Операторы в программе на языке Pascal отделяются друг от друга.

- А) апострофом
- Б) двоеточием;
- В) пробелом;
- Г) запятой;
- Д) точкой с запятой.

20. Логическое выражение $(X \geq A) \text{ AND } (X \leq B)$ истинно при...

- А) x принадлежит отрезку $[A, B]$;
- Б) x принадлежит объединению интервалов $[A, \text{бесконечность}]$ и $[B, \text{бесконечность}]$;
- В) x принадлежит объединению интервалов $[\text{минус бесконечность}, A]$ и $[B, \text{плюс бесконечность}]$;
- Г) x принадлежит объединению интервалов $[A, \text{бесконечность}]$ и $[\text{минус бесконечность}, B]$;
- Д) x принадлежит пересечению интервалов $[\text{минус бесконечность}, A]$ и $[B, \text{плюс бесконечность}]$;

21. Фактические параметры процедуры.

- А) описываются в ее заголовке;
- Б) указываются при описании данных в программе;
- В) перечисляются при ее вызове;
- Г) нигде не указываются;
- Д) указываются при описании промежуточных переменных процедуры.

22. В программе вычисления суммы элементов арифметической прогрессии (при известном ее первом члене, разности и количестве ее членов).

```
Program ArifPro;
Var a, d, s: real; n: integer;
Begin readln(a, d, n); s:= ; writeln(s) End.
```

в операторе присваивания не указано арифметическое выражение. Оно может быть записано как:

- А) $a \cdot n/2 + d \cdot (n-1) \cdot n/2$;
- Б) $a \cdot (n + d \cdot (n-1)) \cdot n/2$;
- В) $a + d \cdot (n-1) \cdot n/2$;
- Г) $a \cdot n + d \cdot (n-1) \cdot n/2$;
- Д) $a/n + d \cdot (n-1) \cdot n/2$.

23. Дана программа.

```

Program T21; Var X: Integer;
Begin Readln(X);
If X MOD 2=0 Then Writeln('ДА')
Else Writeln('НЕТ') End.

```

При каком значении X будет получен ответ “ДА”?

- А) 7;
- Б) 4;
- В) 15;
- Г) 3;
- Д) 21.

24. Программа, написанная на языке Pascal, может содержать несколько разделов. Какой из разделов должен обязательно быть в программе.

- А) тело программы;
- Б) раздел объявления констант;
- В) раздел объявления переменных;
- Г) заголовок.

25. Программа, написанная на языке Pascal, может содержать несколько разделов. С какого специального слова начинается раздел объявления переменных.

- А) VAR;
- Б) CONST;
- В) BEGIN;
- Г) PROGRAM.

26. Программа, написанная на языке Pascal, может содержать несколько разделов. С какого специального слова начинается раздел объявления постоянных (констант).

- А) VAR;
- Б) PROGRAM;
- В) BEGIN;
- Г) CONST.

27. Программа, написанная на языке Pascal, может содержать несколько разделов. С какого специального слова начинается тело программы.

- А) VAR;
- Б) PROGRAM;
- В) BEGIN;
- Г) CONST.

28. При объявлении каждая переменная получает тип. Как обозначается в языке Pascal целый тип.

- А) INTEGER;
- Б) REAL;
- В) STRING;
- Г) ARRAY.

29. Для записи выражений в языке Pascal используют знаки операций. Укажите знаки для операций «сложение» и «вычитание» соответственно.

- А) + и - ;
- Б) - и + ;
- В) * и / ;
- Г) / и * ;
- Д) Div и Mod;

30. Для записи выражений в языке используют знаки операций. Укажите знак для операции «умножение».

- А) + ;
- Б)- С). * ;

В) / ;
 Г) Div.

31. Для записи выражений в языке Pascal используют знаки операций. Укажите знак для операции «деление вещественных чисел».

А) +;
 Б) - ;
 В) * ;

32. К арифметическим операциям не относится знак.

А) «+», «-»;
 Б) «=»;
 В) «*», «/».

33. Укажите оператор ввода с клавиатуры.

А) READ;
 Б) WRITE;
 В) BEGIN .. END;
 Г) IF .. THEN;
 Д) REPEAT ..UNTIL.

34. Укажите функцию квадрата числа.

А) SQR(X)
 Б) SQRT(X)
 В) ABS(X)
 Г) SIN(X)
 Д) COS(X)

35. Укажите функцию извлечения квадратного корня из числа.

А) SQR(X)
 Б) SQRT(X)
 В) ABS(X)
 Г) SIN(X)
 Д) COS(X)

36. Укажите оператор вывода на экран.

А) READ
 Б) WRITE
 В) BEGIN .. END
 Г) IF .. THEN
 Д) REPEAT ..UNTIL

1) Выполнить практическое задание

Простейшие алгоритмы (Приложение А)

Структура программы на языке паскаль (Приложение Б)

Программирование алгоритмов (Приложение В)

Критерии оценивания знаний и умений при проведении промежуточной аттестации:

Оценка «**отлично**» – обучающийся обладает системными теоретическими знаниями (знает методику выполнения практических навыков, показания и противопоказания, возможные осложнения, нормативы и проч.), без ошибок самостоятельно демонстрирует выполнение практических умений,

Оценка «**хорошо**»— обучающийся обладает теоретическими знаниями (знает методику выполнения практических навыков, показания и противопоказания, возможные осложнения, нормативы и проч.), самостоятельно демонстрирует выполнение практических умений, допуская некоторые неточности (малосущественные ошибки), которые самостоятельно обнаруживает и быстро исправляет,

Оценка «**удовлетворительно**» — обучающийся обладает удовлетворительными теоретическими знаниями (знает основные положения методики выполнения практических навыков, показания и противопоказания, возможные осложнения, нормативы и проч.), демонстрирует выполнение практических умений, допуская некоторые ошибки, которые может исправить при коррекции их преподавателем,

Оценка «**неудовлетворительно**» — обучающийся не обладает достаточным уровнем теоретических знаний (не знает методики выполнения практических навыков, показаний и противопоказаний, возможных осложнений, нормативы и проч.) и/или не может самостоятельно продемонстрировать практические умения или выполняет их, допуская грубые ошибки.

Критерии оценивания (за правильный ответ дается 1 балл)

от 0 % до 70 % включительно – оценка «**неудовлетворительно**»;

от 71% до 80% включительно – оценка «**удовлетворительно**»;

от 81 % до 90 % включительно – оценка «**хорошо**»;

от 91 % до 100 % включительно – оценка «**отлично**»

Приложение А

Создайте простейшие алгоритмы в соответствии с приведенными заданиями ниже.

Алгоритмы



— это понятное и точное предписание исполнителю совершить последовательность действий, направленных на решение поставленной задачи. Алгоритмы, предназначенные для выполнения компьютерами, называются программами. Набор правил записи компьютерной программы называется алгоритмическим языком (или языком программирования).

Задание 1. Нарисовать прямоугольник со сторонами 50 на 90.

Сохранить как Задание_1Черепаша. .

Сделайте скриншот и сохраните в файле Фамилия_Практика

Задание 2. Написать программу построения квадрата со стороной 110 точек.

```

1 использовать Черепаша
2 алг квадрат
3 нач
4 . вперед (110)
5 . вправо (90)
6 . вперед (110)
7 . вправо (90)
8 . вперед (110)
9 . вправо (90)
10 . вперед (110)
11 . вправо (90)
12 .
13 кон

```

Недостатки: Многократно повторяются одни и те же команды; ограничение при рисовании сложных фигур.

Изменить программу так, чтобы исключить повторение команд.

Используйте команду Цикл «N раз».

```

1 использовать Черепаша
2 алг квадрат
3 нач
4 . нц 4 раз
5 . . вперед (110)
6 . . вправо (90)
7 . кц
8 кон

```

Использование цикла значительно сокращает программу, однако для того, чтобы построить еще один квадрат, необходимо вновь создавать цикл.

Произвольный прямоугольник со стороной 90 и x:

```

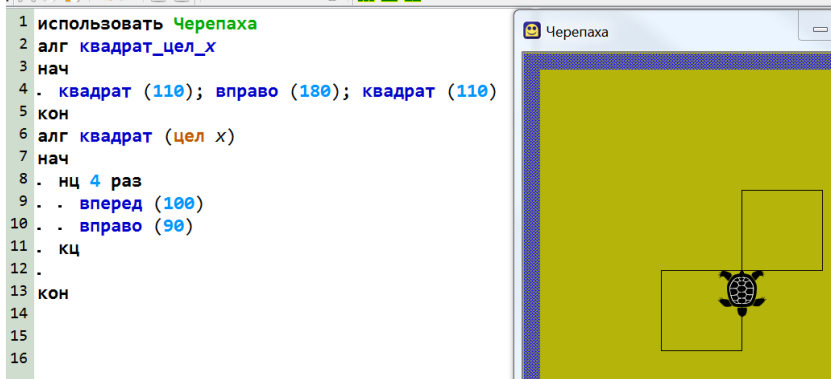
1 использовать Черепаха
2 алг квадрат_цел_x
3 нач
4 . цел x
5 . ввод x
6 . нц 4 раз
7 . . вперед (x)
8 . . вправо (90)
9 . кц
10 кон

```

Использование переменной-параметра, передающего размер стороны квадрата, позволит не только создавать неограниченное количество фигур, но и многократно менять размеры квадрата.

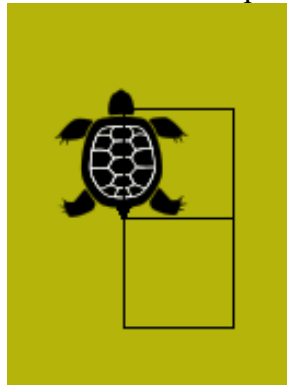
Сделайте скриншот и сохраните в файле Фамилия_Практика

Задание 3. Измените программу так, чтобы симметрично построить еще один квадрат:
Одно из решений:

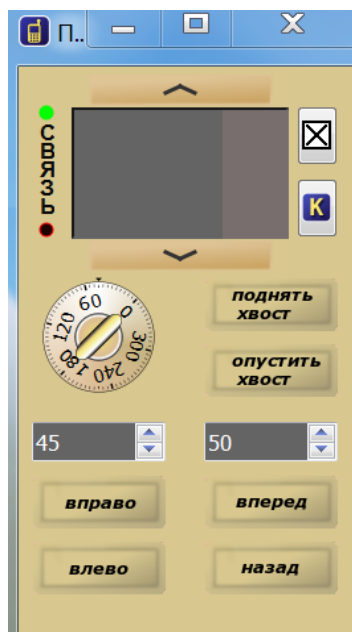


Сделайте скриншот и сохраните в файле Фамилия_Практика

Задание 4. Написать программу для выполнения рисунка:



Если возникают затруднения, то воспользуйтесь пультом управления черепашкой: Миры → Черепаха-пульт:



Для переноса программы из пульта в область поля Кумира воспользуйтесь кнопкой на



пульте

Одно из решений:

```

1 использовать Черепаша
2 алг
3 нач
4 . квадрат
5 . вперед (50)
6 . квадрат
7 кон
8 алг квадрат
9 нач
10 . нц 4 раз
11 . . вперед (50)
12 . . вправо (90)
13 . кц
14 кон

```

Приложение Б

Создайте структуру программы на языке Pascal, исполнитель чертежник в среде Pascal ABC.

Исполнитель Чертежник в среде Паскаль ABC

Команды исполнителя Чертежник содержатся в модуле Drawman:

ToPoint(x,y)– перемещает перо Чертежника в точку (x,y);

OnVector(a,b)– перемещает перо Чертежника на вектор (a,b);

PenUp – поднимает перо Чертежника;

PenDown – опускает перо Чертежника.

Для вызова задания для исполнителя Чертежник используется следующий шаблон программы:

```
uses Drawman;

begin
  Task('a1');
end.
```

В конце программы перо Чертежника должно быть поднято и находиться в начале координат.

Здесь Task - процедура, содержащаяся в модуле Drawman и вызывающая задание с указанным именем.

Имеются следующие группы заданий для исполнителя Чертежник:

a – вводные задания;

c – цикл с параметром;

cc – вложенные циклы;

p – процедуры без параметров;

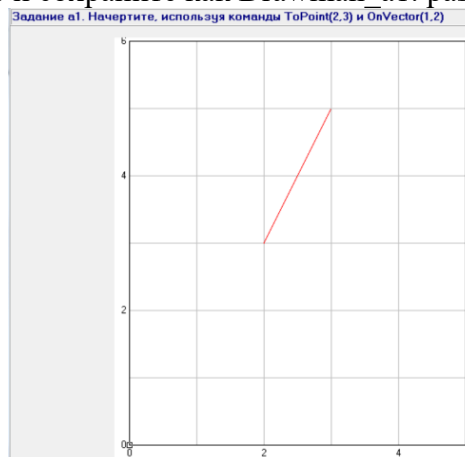
pp – процедуры с параметрами.

Для создания произвольного поля размера 20 x 30 используется процедура StandardField без параметров, а для создания поля размера N x M - процедура Field(N,M).

Исполнитель Чертежник предназначен для построения рисунков и чертежей на плоскости с координатами. Чертежник имеет перо, которое он может поднимать, опускать и перемещать. При перемещении опущенного пера за ним остается след.

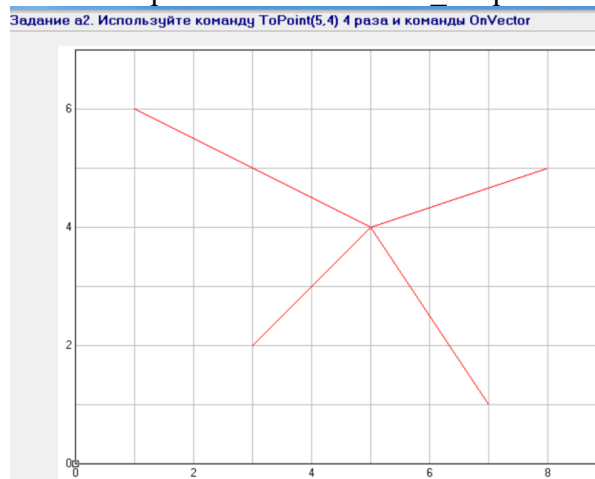
Маленький квадрат изображает Чертежника, красным цветом изображены отрезки, которые надо нарисовать, а синим - уже нарисованные Чертежником отрезки. Когда перо Чертежника опущено, он изображается квадратом меньшего размера.

Задание 1. Выполните a1 и сохраните как Drawman_a1.pas.



В конце программы перо Чертежника должно быть поднято и находиться в начале координат.

Задание 2. Выполните a2 и сохраните как Drawman_a2.pas.

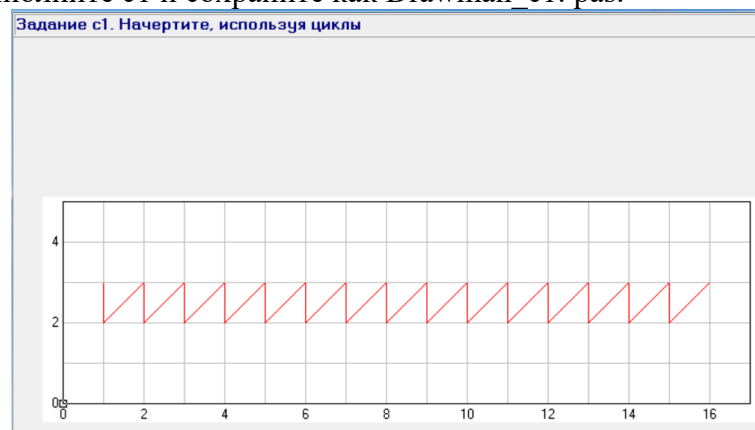


В конце программы перо Чертежника должно быть поднято и находиться в начале координат.

Например:

```
uses Drawman;
begin
  Task('a2');
  PenUp;
  ToPoint(5,4);
  PenDown;
  OnVector(-4,2);
  PenUp;
  ToPoint(5,4);
  PenDown;
  OnVector(-2,-2);
  PenUp;
  ToPoint(5,4);
  PenDown;
  OnVector(3,1);
  PenUp;
  ToPoint(5,4);
  PenDown;
  OnVector(2,-3);
  PenUp;
  ToPoint(0,0);
end.
```

Задание 3. Выполните c1 и сохраните как Drawman_c1.pas.



Например:

```
uses Drawman;  
var  
    i:integer;  
begin  
    Task('c1');  
    begin  
        PenUp;ToPoint(1,3);PenDown;  
        for i:=1 to 15 do  
            begin  
                OnVector(0,-1);OnVector(1,1);  
            end;  
        end;  
        PenUp; ToPoint(0,0);  
    end.
```

Сделайте скриншот (кнопка клавиатуры PrtSc) выполненного задания в текстовом (Word) файле Фамилия_Практика.

Задание 4. Выполните p2 и сохраните как Drawman_p2. pas.



Приложение В

Программирование алгоритмов, создайте алгоритм с помощью операторов присваивания, ввода и вывода.

1. Основные сведения

Программы с линейной структурой состоят из операторов присваивания, ввода и вывода.

Основные операторы

Основным элементарным действием в вычислительных алгоритмах является присваивание значения переменной величине. Если значение константы определено видом ее записи, то переменная величина получает конкретное значение только в результате присваивания, которое может осуществляться двумя способами: с помощью команды присваивания и с помощью команды ввода.

Оператор присваивания : - это основной оператор любого языка программирования. Данный оператор позволяет поместить определенное значение в необходимую вам переменную.

Раздел описания переменных начинается со слова `Var` , за которым идет список имен одностепенных переменных через запятую. В языке Паскаль существует два числовых типа величин: вещественный (`Real`) и целый (`Integer`). Имена переменных составляются из латинских букв и цифр, причем первым символом - буква.

Практически любая программа кроме всего прочего показывает на экране какие-нибудь числа или слова. Оператор, с которым мы познакомимся, приказывает компьютеру изобразить на экране монитора ту или иную информацию, состоящую из символов. Эту задачу выполняет процедура `Write`. То, что нужно изобразить, положено заключать в круглые скобки.

Раздел операторов – основная часть программы. Начало и конец раздела отмечаются служебными словами `BEGIN` и `END`, которые являются операторными скобками. Между этими словами помещаются все команды алгоритма, записанные на языке Паскаль (операторы). Разделителем операторов является точка с запятой. В самом конце программы ставится точка.

Ввод исходных данных с клавиатуры происходит по оператору `READ` или `READLN` (читать). Оператор `READLN` отличается от `READ` только тем, что после ввода данных курсор перемещается в начало новой строки.

Задание 1. Даны катеты прямоугольного треугольника a и b . Найти его площадь $S=0,5*a*b$ и периметр $P=a+b+c = a + b + \sqrt{(a^2 + b^2)}$. Осуществите ввод чисел 3 и 4. Сохранить как `Program_n1`.

Например:

```
Program n1;
Uses Crt;
Var a,b,s,p:Real;
begin
  ClrScr;
  Write('Введи сторону A = '); Readln(a);
  Write('Введи сторону B = '); Readln(b);
  S:=0.5*a*b;
  P:=a+b+sqrt (sqr (a)+sqr (b) );
  Writeln(' S = ',S: 10: 5);
  Writeln(' P = ',P: 10: 5);
end.
```

`Writeln(' P = ',P: 10: 5)` - Выдаст в консоль значение P , всего 10 знаков, из них 5 знаков после десятичной запятой (точки)

Сделайте скриншот (кнопка клавиатуры `PrtSc`) выполненного задания в текстовом (`Word`) файле `Фамилия_Практика`.

Задание 2. Составьте программу вычисления площади треугольника по формуле Герона

$S = \sqrt{p(p-a)(p-b)(p-c)}$, где полупериметр вычисляется по формуле $p = \frac{a+b+c}{2}$, если даны длины сторон треугольника. Подставить следующие варианты сторон 4,5,6; 3,4,5. Сохранить как Program_n2. Сделайте скриншот (кнопка клавиатуры PrtSc) выполненного задания в текстовом (Word) файле Фамилия_Практика.

```
Program n2;
Uses Crt;
Var a,b,c,s,p:Real;
begin
  ClrScr;
  Write('Введи сторону A = '); Readln(a);
  Write('Введи сторону B = '); Readln(b);
  Write('Введи сторону C = '); Readln(c);
  P:=(a+b+c)/2;
  S:=sqrt(p*(p-a)*(p-b)*(p-c));
  Writeln(' S = ',S: 10: 5);
end.
```