

Департамент образования, культуры и спорта Ненецкого автономного округа
государственное бюджетное профессиональное образовательное учреждение
Ненецкого автономного округа
«Ненецкий аграрно-экономический техникум имени В.Г. Волкова» (ГБПОУ
НАО «Ненецкий аграрно-экономический техникум имени В.Г. Волкова»)

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Нарьян-Мар

2024

Рабочая программа учебной дисциплины ОП.04 «Основы алгоритмизации и программирования» разработана на основе примерной программы учебной дисциплины «Основы алгоритмизации и программирования» для профессий начального профессионального образования и специальностей среднего профессионального образования, рекомендованной Федеральным государственным автономным учреждением «Федеральный институт развития образования» (ФГАУ «ФИРО», протокол № 3 от 21 июля 2015 г.), в соответствии с учебным планом ГБПОУ НАО «Ненецкий аграрно-экономический техникум имени В.Г. Волкова» по специальности среднего профессионального образования 09.02.07 Информационные системы и программирование, входящей в состав укрупнённой группы специальностей 09.00.00 Информатика и вычислительные системы.

Организация-разработчик: ГБПОУ НАО «Ненецкий аграрно-экономический техникум имени В.Г. Волкова»

Разработчик:

Кожевин Владимир Григорьевич, преподаватель

Рассмотрена и одобрена к утверждению на заседании предметно – цикловой комиссий естественнонаучных дисциплин ГБПОУ НАО «Ненецкий аграрно – экономический техникум имени В.Г. Волкова»

Заключение предметно – цикловой комиссии естественнонаучных дисциплин № 9 от «24» мая 2024 г.

Председатель ПЦК:  / Кудляк О.А. /

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И ПРИМЕРНОЕ СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	6
3. УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	21
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	23

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

Операционные системы

1.1. Область применения программы

Рабочая программа учебной дисциплины является частью основной образовательной программы в соответствии с ФГОС СПО по специальности среднего профессионального образования 09.02.07 Информационные системы и программирование, укрупнённой группы профессий и специальностей 09.00.00 Информатика и вычислительная техника.

1.2. Место дисциплины в структуре основной профессиональной образовательной программы: учебная дисциплина «Операционные системы и среды» принадлежит к общепрофессиональному циклу.

1.3. Цели и задачи дисциплины – требования к результатам освоения дисциплины:

Цель: освоение теоретических знаний об основах алгоритмизации; приобретение умений применять эти знания в профессиональной деятельности; формированию необходимых компетенций.

В результате освоения дисциплины обучающийся должен уметь:

- Разрабатывать алгоритмы для конкретных задач.
- Использовать программы для графического отображения алгоритмов.
- Определять сложность работы алгоритмов.
- Работать в среде программирования.
- Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.
- Оформлять код программы в соответствии со стандартом кодирования.
- Выполнять проверку, отладку кода программы.

В результате освоения дисциплины обучающийся должен знать:

- Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.
- Эволюцию языков программирования, их классификацию, понятие системы программирования.
- Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;
- Подпрограммы, составление библиотек подпрограмм.
- Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения;
- Объектно-ориентированная модель программирования, понятие классов и объектов, их свойств и методов;
- Общие сведения о файлах, определение файлового типа, спецификация файла;
- Стандартные процедуры и функции обработки файлов.
- В результате освоения дисциплины обучающийся осваивает элементы компетенций:

общих:

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 со коллегами, руководством, клиентами.

ОК 9. Использовать информационные технологии в профессиональной деятельности.

профессиональных:

ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств.

ПК 3.2. Выполнять измерение характеристик компонент программного продукта для определения соответствия заданным критериям.

ПК 5.4. Производить разработку модулей информационной системы в соответствии с техническим заданием.

ПК 6.2. Выполнять исправление ошибок в программном коде информационной системы.

1.4. Рекомендуемое количество часов на освоение программы дисциплины:

максимальной учебной нагрузки обучающегося 188 часов, в том числе: обязательной аудиторной учебной нагрузки обучающегося 160 часа; самостоятельной работы обучающегося 10 часов.

2. СТРУКТУРА И ПРИМЕРНОЕ СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Максимальная учебная нагрузка (всего)	188
Обязательная аудиторная учебная нагрузка (всего)	160
в том числе:	
Теоретическое обучение	90
практические занятия	70
Самостоятельная работа обучающегося (всего)	10
Итоговая аттестация в форме экзамена	

2.2 Содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
Раздел 1 Введение в программирование			
Тема 1.1 Основы алгоритмизации	Объем аудиторной нагрузки		
	1. Алгоритмы. Свойства алгоритмов.	2	2
	2. Способы описания алгоритмов.	2	
	3. Основные конструкции алгоритмического языка: линейный алгоритм, ветвление, цикл.	2	
	4. Этапы решения задач: постановка задачи, создание модели, алгоритм, анализ результатов.	2	
	5. Правила постановки задачи. Модель: входные и выходные параметры, соотношение между ними	2	
	Практические занятия		
	1. Составление простых алгоритмов	2	2
	2. Составление циклических алгоритмов	2	
	3. Трассировка	2	
	Самостоятельная работа обучающихся		
	1. Составление конспекта на тему «История развития термина алгоритма»	2	
Тема 1.2 Языки программирования	Объем аудиторной нагрузки		
	1. Развитие языков программирования. Обзор языков программирования. Области применения языков программирования.	2	2
	2. Стандарты языков программирования. Среда проектирования. Компиляторы и интерпретаторы	2	
	3. Жизненный цикл программы. Основные этапы решения задач на компьютере	2	2
	Практические занятия		
	1. Основные этапы решения задач	2	2

	2. Внутренне представление данных в памяти компьютера	2	
	Самостоятельная работа обучающихся		
	1. Эволюция языков программирования	2	
Тема 1.3 Типы данных	Объем аудиторной нагрузки		
	1. Переменные и константы	2	
	2. Внутренне представление данных в памяти компьютера	2	
	3. Типы данных. Простые типы данных.	2	
	4. Производные типы данных. Структурированные типы данных	2	
	Практические занятия		
	1. Составление программ линейной структуры	2	
	2. Сдвиг	2	
	Самостоятельная работа обучающихся		
	1. Трассировка программы	2	
Раздел 2 Основные конструкции языков программирования			
Тема 2.1 Операторы языка программирования	Объем аудиторной нагрузки		
	1. Основные понятия языка. Алфавит, служебные слова, константы, переменные, имена, типы данных.	2	
	2. Встроенные функции языка. Понятие выражений. Типы выражений	2	
	3. Структура программы. Операции и выражения. Правила формирования и вычисления выражений.	2	
	4. Виды операторов языка. Элементарный ввод и вывод	2	
	5. Логические операторы	2	
	6. Операторы присваивания, условного и безусловного переходов. Оператор выбора	2	
	7. Идентичность и совместимость типов данных при присваивании. Условный оператор	2	
	8. Логические функции в условных операторах if, elif.	2	
	9. Оператор in	2	
	10.Сплит, джоин	2	

	11. Break, continue	2	
	12. Форматированный вывод	2	
	Практические занятия		
	1. Именованное присваивание	2	
	2. Булевы функции	2	
	3. Множественное присваивание.	2	
	4. Типы данных и преобразование	2	
	5. Решение задач с применением оператора if, elif	2	
	Самостоятельная работа обучающихся		
	1. Счетчики	2	
Тема 2.2 Циклы	Объем аудиторной нагрузки		
	1. Цикл for	2	
	2. Цикл с предусловием	2	
	3. Цикл с постусловием	2	
	4. Цикл с параметром. Вложенные циклы	2	
	Практические занятия		
	1. Выбор цикла для решения поставленной задачи	2	
	2. Решение задач с применением вложенных циклов	2	
	3. Однострочный цикл	2	
	Самостоятельная работа обучающихся		
	1. Прерывание циклов	2	
Тема 2.3 Множественный тип данных	Объем аудиторной нагрузки		
	1. Множества	2	
	2. Списки	2	
	3. Строки	2	
	4. Кортежи	2	
	5. Словари	2	
	6. Вложенные списки	2	
	7. Срезы	2	

	8. Преобразование типов. Методы	2	
	Практические занятия		
	1. Решение задач на тему множеств	2	
	2. Решение задач на тему списки	2	
	3. Преобразование строки и поиск по строке	2	
	4. Ord, Chr	2	
	5. Решение задач на тему словари	2	
	6. Решение задач на тему срезы	2	
	7. Преобразование из строки и в строку	2	
	8. Ввод и вывод одномерного массива	2	
	9. Работа с одномерным массивом	2	
	10. Ввод и вывод двумерного массива	2	
	11. Работа с двумерным массивом	2	
	Самостоятельная работа обучающихся		
	1. Решение задач на тему кортежи		
Раздел 3 Структурное и модульное программирование			
Тема 3.1 Процедуры и функции	Объем аудиторной нагрузки		
	1. Общие сведения о подпрограммах. Понятие подпрограмм, подпрограммы	2	
	2. Определение и вызов подпрограмм. Описание процедур. Выполнение процедур	2	
	3. Описание функций, их выполнение. Виды параметров в подпрограммах.	2	
	4. Область видимости и время жизни переменной. Механизм передачи параметров.	2	
	Практические занятия		
	1. Способы задания процедур	2	
	2. Способы задания функций	2	
	3. Выбор процедур или функций для решения поставленной задачи	2	
	4. Возвращение переменных	2	
	5. Рекурсия	2	

	Самостоятельная работа обучающихся		
	1. Применение рекурсий для решения задач		
Тема 3.2 ООП	Объем аудиторной нагрузки		
	1. Подключение библиотек	2	
	2. Понятие ООП	2	
	3. Понятие класса	2	
	4. Наследование	2	
	5. Библиотеки. Создание	2	
	Практические занятия		
	1. Математические и иные библиотеки	2	
	2. Создание классов	2	
	3. Решение задач на тему классов	2	
	4. Решение задач на тему наследование	2	
	5. Работа с файлами		
	6. Ввод и вывод данных из файлов		
	7. Графические библиотеки		
	Самостоятельная работа обучающихся		
	1. Работа с графическими библиотеками		
Промежуточная аттестация		18	
Всего		188	

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

Реализация учебной дисциплины требует наличия Лаборатория программного обеспечения и сопровождения компьютерных систем.

Компьютерные классы оснащены компьютерами с обязательным наличием стационарного проектора или интерактивной доски для проведения следующих видов занятий: комбинаторных занятий, лабораторно-практических занятий, семинаров, дидактических игр.

В состав программных средств должны входить:

- операционная система WINDOWS, Microsoft Office;
- среда программирования Python;
- компилятор с языка программирования;
- браузер;

Средства обучения учебного кабинета:

– учебно-методические указания по выполнению лабораторных работ;

- видео-уроки;

Оборудование кабинета и рабочих мест кабинета:

- Ученические столы;
- Стулья ученические;
- Классная доска;
- Шкафы для хранения пособий;
- Компьютерные столы.

3.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы.

Основные источники:

1. Голицына О.Л. Основы алгоритмизации и программирования: Учебное пособие / О.Л. Голицына. - М.:ФОРУМ : ИНФРА-М, 2006. - 432с.
2. Семакин И.Г. Основы программирования: Учебник / И.Г. Семакин. - М. :Издательский центр «Академия», 2008. - 432с.
3. Основы алгоритмизации и программирования / В.Д. Колдаев. - учебное пособие. - М : ИНФРА-М, 2006. - 400с.
4. Основы построения автоматизированных информационных систем / Н.З. Емельянова. - Учебное пособие. - М. : ИНФРА-М, 2007. - 416с.

Дополнительные источники:

1. Семакин И.Г., Шестаков А.П. Основы программирования:

- Учебник для сред. проф. образования / И.Г. Семакин, А.П. Шестаков. — М.: Издательский центр «Академия», 2008. - 400 с. (Допущено Министерством образования и науки Российской Федерации)
2. Златопольский .М. Сборник задач по программированию. 2-е изд., пере- раб. и доп./ Д. М. Златопольский. - СПб.: БХВ-Петербург, 2007
 3. Основы программирования [Электронный ресурс]/ (WWW.apet.ru/moodle). Проверено 19.09.2016
 4. Википедия [Электронный ресурс]/(WWW.wikipedia.org).
 5. Вычислительные методы и программирование [Электронный ресурс] / Электронные данные. Режим доступа: <http://num-meth.srcc.msu.ru/>, скачивание в формате PDF (Дата обращения: (04.02.2017 г.).

3.3. Организация образовательного процесса

В ходе преподавания учебной дисциплины осуществляются межпредметные связи с профессиональными модулями: ПМ.02 Осуществление интеграции программных модулей., ПМ03 Ревьюирование программных продуктов., ПМ 05 Проектирование и разработка информационных систем, ПМ 06 О Сопровождение информационных систем..

3.4. Кадровое обеспечение образовательного процесса

Требования к квалификации педагогических кадров, обеспечивающих обучение по междисциплинарному курсу (курсам): наличие высшего профессионального образования, соответствующего профилю преподаваемого модуля. Педагогические работники получают дополнительное профессиональное образование по программам повышения квалификации, в том числе в форме стажировки в организациях, направление деятельности которых соответствует области профессиональной деятельности.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения учебной дисциплины осуществляется преподавателем в процессе проведения практических занятий и лабораторных работ, тестирования, а также выполнения обучающимися индивидуальных заданий, проектов, исследований.

Результаты обучения	Критерии оценки	Формы и методы оценки
Перечень знаний, осваиваемых в рамках дисциплины	Характеристики демонстрируемых знаний	Чем и как проверяется
В результате освоения дисциплины обучающийся должен знать: - Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка решения ситуационных задач
Эволюцию языков программирования, их классификацию, понятие системы программирования.	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка решения ситуационных задач
Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка решения ситуационных задач
Подпрограммы, составление библиотек подпрограмм.	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка решения ситуационных задач

Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения.	90-100 % правильных ответов – «5»; 70- 89% правильных ответов – «4»; 50-69 % правильных ответов – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка решения ситуационных задач
В результате освоения дисциплины обучающийся должен уметь: - Разрабатывать алгоритмы для конкретных задач.	90-100 % правильных ответов и выполненных действий – «5»; 70- 89% правильных ответов и выполненных действий – «4»; 50-69 % правильных ответов и выполненных действий – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка результатов выполнения практических работ, практических заданий по самостоятельной работе
Использовать программы для графического отображения алгоритмов.	90-100 % правильных ответов и выполненных действий – «5»; 70- 89% правильных ответов и выполненных действий – «4»; 50-69 % правильных ответов и выполненных действий – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка результатов выполнения практических работ, практических заданий по самостоятельной работе
Определять сложность работы алгоритмов.	90-100 % правильных ответов и выполненных действий – «5»; 70- 89% правильных ответов и выполненных действий – «4»; 50-69 % правильных ответов и выполненных действий – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка результатов выполнения практических работ, практических заданий по самостоятельной работе

Работать в среде программирования.	90-100 % правильных ответов и выполненных действий – «5»; 70- 89% правильных ответов и выполненных действий – «4»; 50-69 % правильных ответов и выполненных действий – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка результатов выполнения практических работ, практических заданий по самостоятельной работе
Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.	90-100 % правильных ответов и выполненных действий – «5»; 70- 89% правильных ответов и выполненных действий – «4»; 50-69 % правильных ответов и выполненных действий – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка результатов выполнения практических работ, практических заданий по самостоятельной работе
Оформлять код программы в соответствии со стандартом кодирования	90-100 % правильных ответов и выполненных действий – «5»; 70- 89% правильных ответов и выполненных действий – «4»; 50-69 % правильных ответов и выполненных действий – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка результатов выполнения практических работ, практических заданий по самостоятельной работе
Выполнять проверку, отладку кода программы.	90-100 % правильных ответов и выполненных действий – «5»; 70- 89% правильных ответов и выполненных действий – «4»; 50-69 % правильных ответов и выполненных действий – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка результатов выполнения практических работ, практических заданий по самостоятельной работе

Выполнять проверку, отладку кода программы.	90-100 % правильных ответов и выполненных действий – «5»; 70- 89% правильных ответов и выполненных действий – «4»; 50-69 % правильных ответов и выполненных действий – «3»; менее 50 % - «2»	устный опрос, тестирование, оценка результатов выполнения практических работ, практических заданий по самостоятельной работе
---	--	---

Контроль формируемых профессиональных и общих компетенций

Формируемые профессиональные и общие компетенции	Формы и методы контроля и оценки результатов обучения
ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств.	Экспертная оценка результатов экзамена по решению проблемной задачи. Проверка самостоятельной работы обучающихся по всем темам курса.
ПК 3.2. Выполнять измерение характеристик компонент программного продукта для определения соответствия заданным критериям.	Анализ результатов наблюдения за деятельностью студентов в процессе выполнения ими учебных заданий при проведении защит творческих работ, презентаций. Экспертное наблюдение за ходом выполнения практической работы, результаты выполнения практических самостоятельных работ не менее 75%
ПК 5.4. Производить разработку модулей информационной системы в соответствии с техническим заданием.	Анализ результатов наблюдения за деятельностью студентов в малых группах при решении проблемных, нестандартных ситуаций при постановке учебной задачи. Тестирование, позволяющее оценить возможности индивида брать на себя ответственность
ПК 6.2. Выполнять исправление ошибок в программном коде информационной системы.	Экспертное наблюдение за ходом выполнения практической работы, результаты выполнения практических самостоятельных работ не менее 75%
ОК 1. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;	Проверка самостоятельной работы обучающихся, Экспертная оценка результатов выполнения исследовательских работ по учебным элементам Анализ результатов наблюдения (по заданным показателям) за деятельностью студентов в процессе выполнения ими учебных заданий
ОК 2. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач	Проверка самостоятельной работы обучающихся, Экспертная оценка результатов выполнения исследовательских работ по учебным элементам Анализ результатов наблюдения (по

профессиональной деятельности.	заданным показателям) за деятельностью студентов в процессе выполнения ими учебных заданий
ОК 4. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.	Анализ результатов наблюдения за деятельностью студентов в малых группах при решении проблемных, нестандартных ситуаций при постановке учебной задачи. Тестирование, позволяющее оценить возможности индивида брать на себя ответственность
ОК 9. Использовать информационные технологии в профессиональной деятельности.	Проверка самостоятельной работы обучающихся, Экспертная оценка результатов выполнения исследовательских работ по учебным элементам Анализ результатов наблюдения (по заданным показателям) за деятельностью студентов в процессе выполнения ими учебных заданий

Департамент образования, культуры и спорта
Ненецкого автономного округа
государственное бюджетное профессиональное образовательное учреждение
Ненецкого автономного округа
«Ненецкий аграрно-экономический техникум имени В.Г. Волкова»
(ГБПОУ НАО «Ненецкий аграрно-экономический техникум имени В.Г.
Волкова»)

Фонд оценочных средств
учебной дисциплины ОП.04 Основы алгоритмизации и программирования

Нарьян-Мар
2024

Фонд оценочных средств учебной дисциплины ОП.04 Основы алгоритмизации и программирования разработан на основе примерной программы учебной дисциплины «Информатика» для профессий начального профессионального образования и специальностей среднего профессионального образования, рекомендованной Федеральным государственным автономным учреждением «Федеральный институт развития образования» (ФГАУ «ФИРО», протокол № 3 от 21 июля 2015 г.), в соответствии с учебным планом ГБОУ СПО НАО «Ненецкий аграрно-экономический техникум» по специальности среднего профессионального образования 09.02.07 Информационные системы и программирование, входящей в состав укрупнённой группы специальностей 09.00.00. Информатика и вычислительные системы и рабочей программы учебной дисциплины ОП.04 Основы алгоритмизации и программирования.

Организация-разработчик: ГБПОУ НАО «Ненецкий аграрно-экономический техникум имени В.Г. Волкова»

Разработчик:

Кожевин Владимир Григорьевич, преподаватель

Ф.И.О., ученая степень, звание, должность

Ф.И.О., ученая степень, звание, должность

Ф.И.О., ученая степень, звание, должность

Рассмотрена и одобрена к утверждению на заседании предметно – цикловой комиссий естественнонаучных дисциплин ГБПОУ НАО «Ненецкий аграрно – экономический техникум имени В.Г. Волкова»

Заключение предметно – цикловой комиссии естественнонаучных дисциплин № 1 от «20» мая 2024 г.

Председатель ПЦК:  /Кудляк О.А./

Содержание

1. Паспорт фонда оценочных средств	3
1.1. Область применения.....	3
2. Комплект материалов для проведения тестового контроля	4
2.1 Комплект материалов для проведения проверочных работ.....	10
2.2 Комплект материалов для проведения лабораторных работ и практических занятий.....	74
3. Контрольно-оценочные средства для промежуточной аттестации ..	85

1. Паспорт фонда оценочных средств

1.1. Область применения

Фонд оценочных средств (далее - ФОС) предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины ОП.04 Основы алгоритмизации и программирования.

ФОС учебной дисциплины ОП.04 Основы алгоритмизации и программирования включает контрольные материалы для проведения текущего контроля и промежуточной аттестации в форме экзамена.

ФОС учебной дисциплины ОП.04 Основы алгоритмизации и программирования разработан в соответствии с программой подготовки специалистов среднего звена по специальности 09.02.07 Информационные системы и программирование, входящей в состав укрупнённой группы специальностей 09.00.00. Информатика и вычислительная техника и рабочей программы учебной дисциплины ОП.04 Основы алгоритмизации и программирования.

2. Контрольно-оценочные средства для текущего контроля

2.1 Комплект материалов для проведения контрольных работ

ОЦЕНОЧНОЕ СРЕДСТВО № 1

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ТЕСТИРОВАНИЯ

Раздел	Раздел 1.Алгоритм.Исполнители алгоритмов
Тема	Тема 1.1. Алгоритм. Базовые алгоритмические конструкции
Форма контроля	тестирование
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7
	ПК 2.8
2. Компоненты ОК	ОК 1
3. Знания	1,2,3,4

Условия выполнения задания

Аудитория, 20 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

задания выполняются в произвольном порядке

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: 20 мин.

Оборудование и оснащение: ПК

Источники

1. Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с

Номер вопроса	Вопрос	Верный ответ	
1	Понятие "алгоритм" произошло...		
	от имени ученого-математика	1	
	от латинского слова, в переводе означающего "исполнитель"		
	от греческого слова, в переводе означающего "по шагам"		
	от латинского слова, в переводе означающего "программирование"		
2	Какое свойство алгоритма означает, что процесс решения задачи должен представлять последовательность простых шагов?		
	результативность		
	определенность		
	дискретность	1	
	массовость		
3	Какое свойство алгоритма означает, что алгоритм должен приводить к решению за конечное число шагов?		
	результативность	1	
	определенность		
	дискретность		
	массовость		
4	Какое свойство алгоритма означает, что каждое правило алгоритма должно быть четким и однозначным?		
	результативность		
	определенность	1	
	дискретность		

	массовость		
5	Какое свойство алгоритма означает, что он должен разрабатываться в общем виде, для решения целого класса однотипных задач?		
	результативность		
	определенность		
	дискретность		
	массовость	1	
6	Какого способа записи алгоритма не существует?		
	словесного		
	графического		
	естественного	1	
	программного		
7	Алгоритм, в котором та или иная команда выполняется в зависимости от истинности условия...		
	линейный		
	разветвляющийся	1	
	циклический		
8	Графическое представление алгоритма		
	таблица		
	блок-схема	1	
	чертеж		
	иерархия		
9	Какой тип блоков не используется в линейном алгоритме?		

	блок условия	1	
	блок действия		
	блок ввода		
	блок вывода		
10	Выберите условие принадлежности точки x отрезку $[a;b]$		
	$(x \leq b) \text{ OR } (x \geq a)$		
	$(x \leq b) \text{ AND } (x \geq a)$	1	
	$(x \leq a) \text{ OR } (x \geq b)$		
	$(x \leq a) \text{ AND } (x \geq b)$		
11	Выберите условие x лежит вне отрезка $[a;b]$		
	$(x < b) \text{ OR } (x > a)$		
	$(x < b) \text{ AND } (x > a)$		
	$(x < a) \text{ OR } (x > b)$	1	
	$(x \leq a) \text{ AND } (x \geq b)$		
12	Какой логической операции в алгебре множеств соответствует операция объединения множеств?		
	дизъюнкции	1	
	конъюнкции		
	импликации		
	инверсии		
13	Какой логической операции в алгебре множеств соответствует операция пересечения множеств?		
	дизъюнкции		
	конъюнкции	1	
	импликации		

	инверсии		
14	Заголовок цикла определяет...		
	количество повторений цикла	1	
	набор команд для повторения		
	условие печати результатов		
	параметры цикла		
15	Составное высказывание ложно тогда и только тогда, когда оба элементарных высказывания ложны. О какой логической операции идет речь?		
	конъюнкция		
	дизъюнкция	1	
	импликация		
	эквиваленция		
16	Составное высказывание истинно тогда и только тогда, когда оба элементарных высказывания истинны. О какой логической операции идет речь?		
	конъюнкция	1	
	дизъюнкция		
	импликация		
	эквиваленция		
17	Что такое тело цикла?		
	параметр цикла		
	условие выхода из цикла		
	первая строка циклической структуры		
	набор команд для повторения	1	
18	Какое свойство алгоритма не выполняется, когда входными данными являются		

	константы?		
	результативность		
	определенность		
	дискретность		
	массовость	1	
19	Какого вида ошибок при отладке программы не существует?		
	грамматические	1	
	синтаксические		
	логические		
20	Алгоритмическая конструкция выбор используется для...		
	определения условия выхода из цикла		
	перечня констант и переменных		
	организации множественного ветвления	1	
	определения тела цикла		
21	В каком виде циклической структуры тело цикла выполнится хотя бы один раз?		
	цикл с параметром		
	цикл с предусловием		
	цикл с постусловием	1	
22	Для какого из указанных значений числа X истинно выражение $\text{NOT}(X < 2) \text{ AND } (X < 3)$?		
	Ответ: 1		
	Ответ: 2	1	
	Ответ: 3		

	Ответ: 4		
23	Для какого из указанных значений числа X истинно выражение $\text{NOT}(X > 2) \text{ OR } (X > 6)$?		
	Ответ: 2	1	
	Ответ: 3		
	Ответ: 4		
	Ответ: 5		

Пакет преподавателя:

№ вопроса	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Верный ответ	1	3	1	2	4	3	2	2	1	2	3	1	2	1	2	1	4	4	1	3	3	2	1

ОЦЕНОЧНОЕ СРЕДСТВО № 2

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ПРОВЕРОЧНОЙ РАБОТЫ

Раздел	Раздел 1.Алгоритм.Исполнители алгоритмов
Тема	Тема 1.2. Исполнители алгоритмов
Форма контроля	проверочная работа
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1, ОК 2, ОК 3, ОК 4, ОК 9
3. Знания	1

Условия выполнения задания

Аудитория, 20 минут в начале занятия.

Инструкция для студентов

1. Последовательность и условия выполнения задания

задания выполняются в произвольном порядке

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: 20 мин.

Оборудование и оснащение: *нет*

Источники

1. Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

Вариант № 1

Задание 1.

Цепочка из трех бусин, помеченных латинскими буквами, формируется по следующим правилам. В конце цепочки стоит одна из бусин А, В, С. На первом месте одна из бусин В, D, С, которой нет на третьем месте. В середине — одна из бусин А, С, Е, В, не стоящая на первом месте. Какая из перечисленных цепочек создана по этим правилам?

1)CBВ 2)ЕАС 3)BCD 4)BCВ

Задание 2.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

$a=11$

$a=112-a*9$

$v=51-33+a*2$

$a=a-2$

$a=a-v/a+39$

где / - операция целочисленного деления.

Задание 3.

Сколько раз выполнится тело цикла, чему равны значения переменных а и в после выполнения фрагмента алгоритма?

Задание 4.

У исполнителя Утроитель две команды, которым присвоены номера:

1. Вычти 1

2. Умножь на 3

Первая из них уменьшает число на экране на 1, вторая - утраивает его. Запишите порядок команд в алгоритме получения из числа 5 числа 26, содержащем не более 5 команд, указывая лишь номера команд.

Задание 5.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записывается буква, следующая в алфавите после первой буквы исходной цепочки, затем две последние буквы исходной цепочки в обратном порядке и, наконец, первая буква исходной цепочки. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка ТНОС.

Дана цепочка символов КОШ. Какая цепочка символов получится, если к данной цепочке применить алгоритм трижды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы, а затем ко второму результату работы алгоритма)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Вариант 2.

Задание 1.

В формировании цепочки из четырех бусин используются некоторые правила. В конце цепочки стоит одна из бусин Р, N, Т, О. На первом — одна из бусин Р, R, Т, О; которой нет на третьем месте. На третьем месте — одна из бусин О, Р, Т, не стоящая в цепочке последней. Какая из перечисленных цепочек могла быть создана с учетом этих правил?

1)PORT 2)TTTO 3)TTOO 4)OORO

Задание 2.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

а=24

в=18

а=а+в/3+16

в=а+2*в

$$a = v / a + v / 2$$

где / - операция целочисленного деления.

Задание 3.

Сколько раз выполнится тело цикла, чему равны значения переменных a и v после выполнения фрагмента алгоритма?

Задание 4.

У исполнителя Утроитель две команды, которым присвоены номера:

1. Вычти 2

2. Умножь на 3

Первая из НИХ уменьшает число на экране на 2, вторая - утраивает его. Запишите порядок команд в программе получения из 11 числа 13, содержащей не более 5 команд, указывая лишь номера команд.

Задание 5.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записываются первые две буквы исходной цепочки в обратном порядке, затем буква, следующая в алфавите за последней буквой исходной цепочки, и, наконец, исходная цепочка символов, записанная в обратном порядке. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка ОСОНОС.

Дана цепочка символов НИЛ. Какая цепочка символов получится, если к данной цепочке применить алгоритм дважды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Вариант 3.

Задание 1.

Для составления цепочек используются бусины, помеченные буквами А, В, С, D, E. На первом месте в цепочке стоит одна из бусин А, С, E. На втором — любая гласная, если первая буква согласная, и любая согласная, если первая гласная. На третьем месте — одна из бусин С, D, E, не стоящая в цепочке на первом месте. Какая из перечисленных цепочек создана по этим правилам?

1)CBE 2)ADD 3)ECE 4)EAD

Задание 2.

Определить значения переменных a и v после исполнения данного алгоритма, построив трассировочную таблицу.

$$a = 15$$

$$v = 2 * a + 24$$

$$a = v/2 + a$$

$$v = 2 * a + 2 * v + 9$$

$$a = v/a + 2 * a$$

где / - операция целочисленного деления.

Задание 3.

Сколько раз выполнится тело цикла, чему равны значения переменных I и A после выполнения фрагмента алгоритма?

Задание 4.

У исполнителя Калькулятор две команды, которым присвоены номера:

1. Прибавь 2

2. Умножь на 3

Выполняя первую из них, Калькулятор прибавляет к числу на экране 2, а выполняя вторую, утраивает его. Запишите порядок команд в программе получения из 0 числа 28, содержащей не более 6 команд, указывая лишь номера команд.

Задание 5.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записывается буква, предшествующая в алфавите первой букве исходной цепочки, затем исходная цепочка символов, записанная в обратном порядке. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка РНОС.

Дана цепочка символов РОК. Какая цепочка символов получится, если к данной цепочке применить алгоритм трижды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы, а затем ко второму результату работы)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Вариант 4.

Задание 1.

Цепочка из трех бусин формируется по следующим правилам. На первом месте в цепочке стоит одна из бусин А, Б, В. На втором — одна из бусин Б, В, Г. На третьем месте — одна из бусин А, В, Г, не стоящая в цепочке на первом или втором месте. Какая из следующих цепочек создана по этим правилам?

1)АГБ 2)ВАГ 3)БГГ 4)ББГ

Задание 2.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

$a=48$

$b=a/4-5$

$a= a-b$

$b=2*a+37$

$a=b/a+2*a$

где / - операция целочисленного деления.

Задание 3.

Сколько раз выполнится тело цикла, чему равны значения переменных I, A и R после выполнения фрагмента алгоритма?

Задание 4.

Исполнитель Калькулятор имеет только две команды, которым присвоены номера:

1. Вычти 3

2. Умножь на 2

Выполняя команду номер 1, Калькулятор вычитает из числа на экране 3, а выполняя команду номер 2, умножает число на экране на 2. Напишите программу, содержащую не более 5 команд, которая из числа 5 получает число 25. Укажите лишь номера команд.

Задание 5.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записывается буква, предшествующая в алфавите последней букве исходной цепочки, затем первые две буквы исходной цепочки, и наконец, буква, следующая в алфавите за первой буквой исходной цепочки символов. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка МСОТ.

Дана цепочка символов АРТ. Какая цепочка символов получится, если к данной цепочке применить алгоритм дважды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

ОЦЕНОЧНОЕ СРЕДСТВО № 3
КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ПОДГОТОВКИ ДОКЛАДОВ

Раздел	Раздел 2.Язык программирования Pascal
Тема	Тема 2.1. История и классификация языков программирования
Форма контроля	Поисковое задание
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ОК	ОК 1, ОК4, ОК5, ОК8, ОК9
2. Знания	1,2,4

Условия выполнения задания

Аудитория, 20 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

задания выполняются в произвольном порядке

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: 20 мин.

Оборудование и оснащение: ПК, Интернет

Источники

2. *Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.*

Темы докладов:

1. История и классификация языков программирования.
2. Системы программирования.
3. Процедурные и непроцедурные языки программирования.
4. Объектно-ориентированное программирование.
5. Язык программирования Паскаль. История создания. Использование в современности.
6. Биография Никлауса Вирта.

ОЦЕНОЧНОЕ СРЕДСТВО № 4

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ТЕСТИРОВАНИЯ

Раздел	Раздел 2.Язык программирования Pascal
Тема	Тема 2.2. Язык программирования Pascal . Операторы ввода, вывода, присваивания
Форма контроля	тестирование
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1
3. Знания	3

Условия выполнения задания

Аудитория, 20 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

задания выполняются в произвольном порядке

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: 20 мин.

Оборудование и оснащение: ПК

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

Номер вопроса	Вопрос	Верный ответ
1	Языки программирования - это...	
	искусственные языки	1
	естественные языки	
	смешанные языки	
2	Как называется средство для преобразования текстов из языков программирования высокого уровня в машинный код?	
	маршрутизатор	
	транслятор	1
	автокод	
	контроллер	
3	Какой из языков программирования является машинно-зависимым?	
	бейсик	
	паскаль	
	ассемблер	1
	си	
4	Какой язык программирования относится к языкам низкого уровня?	
	ассемблер	1
	бейсик	
	фортран	
	кобол	
5	Какой язык программирования относится к языкам высокого уровня?	
	ассемблер	

	автокод	
	си	1
6	Что такое алфавит языка программирования?	
	фиксированный для данного языка набор символов	1
	буквы, используемые в данном языке	
	правила записи объектов языка	
7	Что такое идентификатор?	
	фиксированный для данного языка набор символов	
	последовательность символов, применяющаяся для имен констант, переменных, функций	1
	средство трансляции в машинный код	
	средство, определяющее тип данных	
8	К какому типу данных относится число 23.56?	
	string	
	integer	
	longint	
	real	1
9	Выберите пример правильного имени переменной:	
	FFG11	1
	6PO	
	АПРЛ	
	FF+HH	

10	Выберите пример неправильного имени переменной:	
	DFR56	
	ИБЛ23	1
	_VFA	
	DH55DF	
11	Выберите идентификатор целого типа данных	
	word	1
	double	
	extended	
	char	
12	Выберите идентификатор логического типа данных	
	word	
	boolean	1
	extended	
	char	
13	Выберите пример описания интервального типа данных	
	type season = (winter,spring,summer,autumn);	
	var color:(red,yellow,green);	
	var fad:=100..200;	1
	var cad:integer;	
14	Чему равен результат операции (47 mod 15)?	
	7.	
	2.	1
	4.	

	10.	
15	Величина, не изменяющая свое значение в процессе выполнения программы называется...	
	константа	1
	переменная	
	функция	
	выражение	
16	Чему равен результат операции $(67 \div 10)$?	
	7.	
	6.	1
	4.	
	10.	
17	Чему равно значение выражения $\text{Frac}(45.56)$?	
	46.	
	45.	
	0.45	
	0.56	1
18	Чему равно значение выражения $\text{Inc}(5,5)$?	
	5.	
	10.	1
	0.	
	55.	
19	Чему равно значение выражения $\text{Dec}(15,10)$?	
	25.	

	10.	
	5.	1
	15.	
20	Чему равно значение выражения $\text{Inc}(20)$?	
	19.	
	10.	
	2.	
	21.	1
21	Чему равно значение выражения $\text{Dec}(15)$?	
	15.	
	14.	1
	16.	
	25.	
22	Функция SQR предназначена для:	
	возведения аргумента в третью степень	
	вычисления квадратного корня из аргумента	
	возведения аргумента во вторую степень	1
	вычисления корня третьей степени из аргумента	
23	С понятием переменная не связано следующее понятие:	
	тип	
	идентификатор	
	объект	1
	значение	

24	Внутреннюю форму представления данных в ЭВМ определяет:	
	идентификатор	
	транслятор	
	тип данных	1
	операция	
25	Раздел описания переменных задается служебным словом:	
	const	
	uses	
	type	
	var	1
26	Тело программы начинается после служебного слова:	
	begin	1
	var	
	uses	
	type	
27	Создателем языка программирования Pascal является:	
	Паскаль	
	Вирт	1
	Лавлейс	
	Куртс	

№	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	26	27
---	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

ВАРИАНТ 1

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{4x^2 + 1}{x - 5}, & \text{если } x < 5, \\ 3x^2 - 2, & \text{если } x \geq 5. \end{cases}$$

ВАРИАНТ 2

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{5x^2 + 2}{x + 4}, & \text{если } x > -4, \\ 3x^2 + 7, & \text{если } x \leq -4 \end{cases}$$

ВАРИАНТ 3

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{7x^2 - 1}{2x + 6}, & \text{если } x < -3, \\ 4x^2 - 5, & \text{если } x \geq -3 \end{cases}$$

ВАРИАНТ 4

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{9x^2 + 5}{3x + 12}, & \text{если } x < -4, \\ 4x^2 - 7, & \text{если } x \geq -4 \end{cases}$$

ОЦЕНОЧНОЕ СРЕДСТВО № 6

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ПРОВЕРОЧНОЙ РАБОТЫ

Раздел	Раздел 2. Язык программирования Pascal
Тема	Тема 2.4. Программирование циклических алгоритмов
Форма контроля	Проверочная работа
Вид контроля	оперативный

Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1,2,3,4,9
3. Знания	3

Условия выполнения задания

Аудитория, 15 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

задания выполняются в произвольном порядке

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: *15 мин.*

Оборудование и оснащение: *нет*

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

Вариант 1

1. Составить программу вычисления суммы и произведения четных чисел из промежутка от 1 до 10.

Вариант 2

2. Составьте программу вывода на экран всех нечетных трехзначных чисел.

Вариант 3

3. Дано целое число N(больше 0). Найти сумму $1+1/2+1/3+...+1/N$ (вещественное число)

Пакет преподавателя

Вариант 1

```

program jjj;
var a: array [1..10] of integer; i,summa,proizv:integer;
begin
for i:=1 to 10 do begin
    Writeln('напишите элемент массива номер ',i); read(a[i]);
    end;
For i:=1 to 10 do begin
    if a[i] mod 2=0 then summa:=summa+a[i] else
    if (a[i]<=10) and (a[i]>=1) then  proizv:=proizv*a[i];
    end;
Writeln(summa,' ',proizv); readln;
end.

```

Вариант 2

```

var i,s: integer;
begin
i := 100;
s := 0;
repeat
s := s + i;
i := i + 5;
until i <= 1000;
writeln(s);
end.

```

Вариант 3

```

var H: real;
    N, i: integer;
begin
    H := 0;
    readln(N);
    for i := 1 to N do
        H := H + 1/i;
    writeln(H:6:4)
end.

```

ОЦЕНОЧНОЕ СРЕДСТВО № 7
КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ОПРОСА

Раздел	Раздел 2.Язык программирования Pascal
Тема	Тема 2.4. Программирование циклических алгоритмов
Форма контроля	Проверочная работа
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, 2.8
2. Компоненты ОК	ОК 1,9
3. Знания	3

Условия выполнения задания

Аудитория, 10 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

студенты отвечают на вопросы

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: *10* мин.

Оборудование и оснащение: *нет*

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

1) Оператор цикла с предусловием, формат команды.

2) Заголовок и тело цикла с предусловием.

- 3) Оператор цикла с параметром, формат команды.
- 4) Заголовок и тело цикла с параметром.
- 5) Шаг изменения счетчика цикла с параметром.
- 6) Использование служебных слов `to` и `downto` в заголовке цикла с параметром.
- 7) Оператор цикла с постусловием, формат команды, условие и тело цикла.
- 8) Организация повтора программы с использованием цикла с постусловием.
- 9) Использование составного оператора в теле цикла.

ОЦЕНОЧНОЕ СРЕДСТВО № 8
КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ТЕСТИРОВАНИЯ

Раздел	Раздел 2. Язык программирования Pascal
Тема	Тема 2.5. Строки и Тема 2.6. Множества
Форма контроля	тестирование
Вид контроля	рубежный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1
3. Знания	1,2,3

Условия выполнения задания

Аудитория, 45 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

студенты отвечают на вопросы

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: 45 мин.

Оборудование и оснащение: ПК

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

Номер вопроса	Вопрос	Верный ответ
1	Выберите правильную форму описания строки s на языке Паскаль.	
	var s:length(s);	
	var s:set of char;	
	var s:string;	1
	type s of string;	
2	Выберите правильную форму описания множества, состоящего из целых чисел.	
	var M:set of integer;	
	var K:set of char;	
	var D:set of string;	
	var T:set of byte;	1
3	Функция, определяющая длину строки s.	
	length(s);	1
	concat(s);	
	pos(s);	
	insert(s);	

4	Назначение функции CHR(N).	
	Определяет по символу порядковый номер	
	Определяет по порядковому номеру символ	1
	Объявляет символьный тип данных	
	Выдает номер первого вхождения символа N в строке s	
5	Назначение функции ORD(C).	
	Определяет по символу порядковый номер	1
	Определяет по порядковому номеру символ	
	Объявляет символьный тип данных	
	Выдает номер первого вхождения символа N в строке s	
6	Назначение функции POS(s1,s).	
	Вставляет подстроку s1 в строку s	
	Удаляет из строки s подстроку s1	
	Ищет первое вхождение подстроки s1 в строке s	1
	Сцепляет строки s1,s	
7	Назначение функции Concat(s1,s).	
	Вставляет подстроку s1 в строку s	
	Удаляет из строки s подстроку s1	
	Ищет первое вхождение подстроки s1 в строке s	
	Сцепляет строки s1,s	1
8	Назначение функции Str(x,s).	
	Вставляет символ x в строку s	
	Удаляет из строки s символ x	
	Ищет первое вхождение подстроки x в строке s	

	Преобразует числовое значение x в строку s	1
9	Назначение функции Val(s,x,error).	
	Преобразует строковое значение s в числовое x	1
	Вставляет символ x в строку s	
	Удаляет из строки s символ x	
	Преобразует числовое значение x в строку s	
10	Определите результат выполнения операции [1,2,5,9]*[1,3,5,7]	
	[1,2,3,5,7,9]	
	[1,2,3]	
	[1,5]	1
	[2,9]	
11	Определите результат выполнения операции [1,2,5,9]+[1,3,5,7]	
	[1,2,3,5,7,9]	1
	[1,2,3]	
	[1,5]	
	[2,9]	
12	Определите результат выполнения операции [1,2,5,9]-[1,3,5,7]	
	[1,2,3,5,7,9]	
	[1,2,3]	
	[1,5]	
	[2,9]	1

13	Определите результат выполнения операции 'a' in ['a','b','c','d']	
	false	
	true	1
	['a']	
	['a','b','c','d']	
14	Операция in, применяемая в Паскале ко множественному типу, используется для...	
	включения нового элемента во множество	
	исключения элемента из множества	
	проверки принадлежности элемента множеству	1
	объединения двух множеств	
15	Определите результат выполнения операции 7 in [1..5]	
	false	1
	true	
	[1..5,7]	
	[1..7]	
16	Что выведется на экран в результате выполнения следующего фрагмента программы? S:='оса'; s1:=copy(s,1,2); k:='новатор'; delete(k,5,3); k1:=concat(s1,k); write(k1);	
	основа	1
	осор	
	новатороса	

	аватар	
17	Что выведется на экран в результате выполнения следующего фрагмента программы? <pre>S:='ЧЭМК'; L:=length(s); write(L);</pre>	
	6.	1
	5.	
	4.	
	3.	
18	Что выведется на экран в результате выполнения следующего фрагмента программы? <pre>S:='HELLO'; k:=pos('L',s); write(k);</pre>	
	1.	
	2.	
	3.	1
	4.	
19	Для получения из строки s:='формальность' строки s:='форма' необходимо использовать команду...	
	insert(s,1,5);	
	copy(s,5,1);	
	delete(s,6,7);	1
	delete(s,1,5);	
20	Для получения из строки s:='огород' строки s:='город' необходимо использовать команду...	
	delete(s,1,5);	

	delete(s,1,1);	1
	delete(s,2,6);	
	delete(s,1,2);	

Пакет преподавателя:

№ вопроса	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Верный ответ	3	4	1	2	1	3	4	4	1	3	1	4	2	3	1	1	1	3	3	2

ОЦЕНОЧНОЕ СРЕДСТВО № 9

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ОПРОСА

Раздел	Раздел 2.Язык программирования Pascal
Тема	Тема 2.7. Массивы
Форма контроля	Фронтальный опрос
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1, 9
3. Знания	3

Условия выполнения задания

Аудитория, 10 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

студенты отвечают на вопросы

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: *10* мин.

Оборудование и оснащение: *ПК*

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

1) Описание одномерного массива.

2) Использование операторов цикла при решении задач обработки массивов.

3) Выделение отдельного элемента одномерного массива.

4) Поиск элементов в массиве, кратных заданному числу (условие кратности). Привести фрагмент программы.

5) Описание двумерного массива.

6) Использование операторов цикла при решении задач обработки массивов.

7) Выделение отдельного элемента двумерного массива.

8) Свойства диагональных элементов квадратной матрицы.

ОЦЕНОЧНОЕ СРЕДСТВО № 10

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ОПРОСА

Раздел	Раздел 2.Язык программирования Pascal
Тема	Тема 2.8. Подпрограммы
Форма контроля	Фронтальный опрос
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1,9
3. Знания	3

Условия выполнения задания

Аудитория, 10 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

студенты отвечают на вопросы

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: *10* мин.

Оборудование и оснащение: ПК

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

1) Понятие подпрограммы, назначение подпрограмм.

2) Виды подпрограмм в языке программирования Pascal.

3) Состав подпрограммы.

- 4) Вызов подпрограммы.
- 5) Функции, описание функции, заголовок функции, тело функции.
- 6) Процедуры, описание процедуры, заголовок процедуры, тело процедуры.
- 7) Параметры-значение и параметры-переменные процедуры.
- 8) Формальные и фактические параметры, работа механизма передачи параметров.

ОЦЕНОЧНОЕ СРЕДСТВО № 11

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ОПРОСА

Раздел	Раздел 2.Язык программирования Pascal
Тема	Тема 2.9. Записи
Форма контроля	Фронтальный опрос
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1,9
3. Знания	3

Условия выполнения задания

Аудитория, 7 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

студенты отвечают на вопросы

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: 7 мин.

Оборудование и оснащение: *ПК*

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

- 1) Структурированный тип запись, определение, назначение.
- 2) Описание типа запись, используемые служебные слова.
- 3) Компоненты записи и их типы.
- 4) Массив записей, описание, назначение.
- 5) Составные имена.
- 6) Оператор присоединения и его область действия.

ОЦЕНОЧНОЕ СРЕДСТВО № 12

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ СОЗДАНИЯ И ЗАЩИТЫ ПРЕЗЕНТАЦИИ

Раздел	Раздел 2.Язык программирования Pascal
Тема	Тема 2.10. Доступ к файлам
Форма контроля	Исследовательское групповое задание
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ОК	ОК 6,7,8
3. Знания	3

Условия выполнения задания

Аудитория, 2 часа.

Инструкция для студентов

1. Последовательность и условия выполнения задания

Студенты работают в группе по 3 человека, создают презентацию, затем защищают ее.

2. Вы можете воспользоваться: *Конспект лекции, Интернет*

3. Максимальное время выполнения задания: 2 часа.

Оборудование и оснащение: *ПК, Интернет*

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

Вопросы, которые необходимо раскрыть при создании презентации:

- 1) Описание файловых переменных.
- 2) Установление связи между программой и файлом на диске.
- 3) Открытие файла для чтения, записи, добавления данных.
- 4) Запись данных в файл, чтение данных из файла.
- 5) Завершение работы с файлом.

ОЦЕНОЧНОЕ СРЕДСТВО № 13

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ПРОВЕРОЧНОЙ РАБОТЫ

Раздел	Раздел 2.Язык программирования Pascal
Тема	Тема 2.11. Программирование графики
Форма контроля	Проверочная работа
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1,2,3,4,9
3. Знания	3

Условия выполнения задания

Аудитория, 15 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

задания выполняются в произвольном порядке

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: 15 мин.

Оборудование и оснащение: *нет*

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

Вариант 1

1. Что означает команда языка Pascal Ellipse(50,100,150,200)
2. Что означает команда языка Pascal SetBrushColor(clRed)
3. Написать команду рисования прямоугольника, заливка зеленым цветом

Вариант 2

1. Что означает команда языка Pascal SetPixel(100,200,clRed)
2. Что означает команда языка Pascal SetPenColor(clGreen)
3. Написать команду рисования эллипса, заливка синим цветом

Вариант 3

1. Что означает команда языка Pascal Rectangle(50,100,250,300)
2. Что означает команда языка Pascal Uses GraphABC
3. Написать команду рисования круга, заливка красным цветом

Вариант 4

1. Что означает команда языка Pascal FloodFill(100,150,clBlue)
2. Что означает команда языка Pascal SetPenWidth(5)
3. Написать команду рисования линии красным цветом

ОЦЕНОЧНОЕ СРЕДСТВО № 14

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ПОДГОТОВКИ ДОКЛАДОВ

Тема 3.1. Язык программирования С

Раздел	Раздел 3 Язык программирования С
Тема	Тема 3.1. Язык программирования С
Форма контроля	Поисковое задание
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ОК	ОК 1,4,8,9
3. Знания	4

Условия выполнения задания

самостоятельная работа студентов

Инструкция для студентов

1. Последовательность и условия выполнения задания *работа выполняется вне урока, возможно групповое выполнение задания*
2. Вы можете воспользоваться *Интернет ресурсами и др. источниками*
3. Максимальное время выполнения задания: 4 часа.

Оборудование и оснащение *доска с мелом или проектор и ПК*

Темы докладов:

1. Язык программирования С. История. Возможности.
2. Биография Д.Ритчи.

ОЦЕНОЧНОЕ СРЕДСТВО № 15

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ КОМПЬЮТЕРНОГО ТЕСТИРОВАНИЯ

Тема 3.2. Основные операторы языка программирования С

Раздел	Раздел 3 Язык программирования С
Тема	Тема 3.2. Основные операторы языка программирования С
Форма контроля	тестирование
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1
3. Знания	4

Условия выполнения задания

компьютерное тестирование 20 минут в конце занятия

Инструкция для студентов

1. Последовательность и условия выполнения задания *все задания выполняются последовательно, самостоятельно.*
2. Вы можете воспользоваться *нет*

3. Максимальное время выполнения задания: 20 мин.

4. Перечень раздаточных и дополнительных материалов *нет*

Оборудование и оснащение: ПК

Источники

Программирование на языке C++: Учебное пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев; Под ред. Л.Г. Гагариной. - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 512 с.

Номер вопроса	Вопрос	Верный ответ
1	Выберите правильно записанное условие проверки числа x на четность на языке СИ:	
	if (x%2=0)	
	if (x%2==2)	
	if (x mod 2 =0)	
	if (x%2==0)	1
2	<pre>int mark; printf("Vvedite ocenku"); scanf("%d",&mark); if (mark == 4) printf("Vi udarnik.\n"); else if (mark > 4) printf("Vi otlichnik!\n"); else printf ("Ychitsya nado.\n");</pre> С клавиатуры вводится 3, какой результат будет на экране?	
	Vi otlichnik!	
	Vi udarnik.	
	Ychitsya nado.	1
3	Значение вещественной переменной нужно вывести следующим образом: выравнивание по левому краю, три знака после запятой. Выберите	

	соответствующую команду.	
	printf("%f5.3",x);	
	printf("%d3",x);	
	printf("%f-8.3",x);	1
	printf("%-3f",x);	
4	Укажите правильно заданную команду для вывода вещественной переменной a.	
	printf("%f",a);	1
	scanf("%d",&a);	
	scanf("%f",&a);	
	printf("%d",a);	
5	&&	
	логическое "И"	1
	логическое "ИЛИ"	
	логическое "НЕ"	
6		
	логическое "И"	
	логическое "ИЛИ"	1
	логическое "НЕ"	
7	!	
	логическое "И"	
	логическое "ИЛИ"	

	логическое "НЕ"	1
8	"не равно" в СИ обозначается	
	==	
	<>	
	><	
	!=	1
9	Чему равен результат <code>row(3,3)%5=</code>	
	4.	
	5.	
	2.	1
	3.	
10	Описание вещественного типа с СИ	
	float	1
	int	
	real	
	char	
11	Какой спецификатор нужно использовать для вывода вещественного числа?	
	%d	
	%f	1
	%c	
	%i	
12	Какой спецификатор нужно использовать для вывода целого числа?	

	%d	1
	%f	
	%c	
	%s	
13	Какой спецификатор нужно использовать для вывода одного символа?	
	%d	
	%f	
	%c	1
	%i	
14	Требуется вывести вещественное число в СИ, так чтобы после запятой было 3 знака. Какой формат нужно использовать?	
	%3d	
	%3f	
	%4.3f	1
	%4.3d	
15	В каком формате появится число на экране при использовании команды: <code>printf("%4.2f",6);</code>	
	6.	
	6.00	1
	6.00000	
	6.000000	
16	Какой оператор в СИ находит остаток от деления?	
	/	
	mod	
	%	1

	&	
17	Оператор получения адреса	
	%	
	&&	
	&	1
	\n	
18	\n в СИ обозначает:	
	горизонтальная табуляция	
	перевод курсора на новую строку	1
	спецификатор целочисленного формата	
	спецификатор вещественного формата	
19	Оператор соотношения "равно" записывается:	
	=	
	:=	
	!=	
	==	1
20	Как записывается логическое "И" в СИ?	
	and	
	&	
	&&	1
	i	
21	Как записывается логическое "ИЛИ" в СИ?	
	ili	

	OR	
		1
22	Как записывается логическое "НЕ" в СИ?	
	!	1
	not	
	&	
23	Язык СИ разработал...	
	Том Куртц	
	Никлаус Вирт	
	Деннис Ритчи	1
	Билл Гейтс	
24	Чему равен результат выполнения операции $15\%3=...$	
	0.	1
	1.	
	5.	
	3.	
25	Какую библиотеку нужно подключить для работы операторов scanf() и printf() в СИ?	
	<math.h>	
	<stdio.h>	1
	<conio.h>	
	<stdlib.h>	

26	Какую библиотеку нужно подключить для работы оператора fabs() в СИ?	
	<math.h>	1
	<stdio.h>	
	<conio.h>	
	<stdlib.h>	
27	Как находится модуль вещественного числа в СИ?	
	mod	
	fabs	1
	abs	
28	Как записать на СИ "корень квадратный из x"?	
	x ^ 0.5	
	power(x,2)	
	pow(x,1/2)	1
	pow(x,y,2)	
29	Оператор форматированного вывода:	
	scanf()	
	print()	
	printf()	1
	write()	
30	Оператор ввода:	
	scanf()	1
	read()	

	printf()	
	input()	
31	Укажите правильно заданную команду для ввода вещественной переменной a.	
	printf("%f",a);	
	scanf("%d",&a);	
	scanf("%f",&a);	1
	scanf("%-f",a);	

Пакет преподавателя:

№ вопроса	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	26	27	28	29	30	31
Верный ответ	4	3	3	1	1	2	3	4	3	1	2	1	3	3	2	3	3	2	4	3	4	2	3	1	2	1	2	3	3	1	3

ОЦЕНОЧНОЕ СРЕДСТВО № 16

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ПРОВЕРОЧНОЙ РАБОТЫ

Тема 3.3. Массивы

Раздел	Раздел 3 Язык программирования C
Тема	Тема 3.3. Массивы
Форма контроля	Проверочная работа
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1,2,3,4,9

3. Знания	4
------------------	---

Условия выполнения задания

Аудитория, 20 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

Создать программу, описать предполагаемый результат

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: *20 мин.*

Оборудование и оснащение: *нет*

Источники

Программирование на языке C++: Учебное пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев; Под ред. Л.Г. Гагариной. - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 512 с.

Вариант 1

Написать программу на языке программирования C. В массиве из 20 целочисленных элементов, заданных случайным образом из отрезка [-15;15] найти максимальный элемент и его номер.

Вариант 2

Написать программу на языке программирования C. В массиве из N(вводится с клавиатуры) целочисленных элементов, заданных случайным образом из отрезка [-5;5] подсчитать количество нечетных положительных элементов, вывести найденные элементы на печать.

Вариант 3

Написать программу на языке программирования C, которая определяет в целочисленной матрице, состоящей из элементов из отрезка [-5;5], номер строки, которая содержит наибольшее количество элементов, равных нулю.

ОЦЕНОЧНОЕ СРЕДСТВО № 17
КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ТЕСТИРОВАНИЯ
(рубежный контроль)

Раздел	Раздел 3 Язык программирования С
Тема	Тема 3.4. Работа со строковым типом Тема 3.5. Подпрограммы Тема 3.6. Программирование с использованием структур Тема 3.7. Доступ к файлам
Форма контроля	тестирование
Вид контроля	рубежный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Компоненты ОК	ОК 1
3. Знания	4

Условия выполнения задания

Аудитория, 45 минут.

Инструкция для студентов

1. Последовательность и условия выполнения задания

Студенты последовательно отвечают на вопросы

2. Вы можете воспользоваться: *нет*

3. Максимальное время выполнения задания: 45 мин.

Оборудование и оснащение: ПК

Источники

Номер вопроса	Вопрос	Верный ответ
1	If (выражение) оператор_1 ;else оператор_2;	
	структура оператора условия	1
	структура цикла с параметром	
	структура оператора множественного выбора	
	структура условного выражения	
2	<pre>int mark; printf("Vvedite ocenku"); scanf("%d",&mark); if (mark == 4) printf("Vi udarnik.\n"); else if (mark > 4) printf("Vi otlichnik!\n"); else printf ("Ychitsya nado.\n");</pre> С клавиатуры вводится 3, какой результат будет на экране?	
	Vi otlichnik!	
	Vi udarnik.	
	Ychitsya nado.	1
3	<pre>int mark; printf("Vvedite ocenku"); scanf("%d",&mark); if (mark == 4) printf("Vi udarnik.\n"); else if (mark > 4) printf("Vi otlichnik!\n"); else printf ("Ychitsya nado.\n");</pre> С клавиатуры вводится 5, какой результат будет на экране?	
	Vi otlichnik!	1
	Vi udarnik.	
	Ychitsya nado.	

4	<pre>int a,b,i; printf("Vvedite a\n"); scanf("%d",&a); printf("Vvedite b\n"); scanf("%d",&b); if(a == 0) printf("Net resheniya.\n"); else printf("Ect reshenie=%2.3f\n",b/a);</pre> С клавиатуры ввели 5 и 10. Какой будет результат?	
	Ect reshenie=0.5	
	Ect reshenie=2	1
	Net resheniya	
5	Оператор «&&» обозначает	
	логическое "И"	1
	логическое "ИЛИ"	
	логическое "НЕ"	
6	Оператор « » обозначает	
	логическое "И"	
	логическое "ИЛИ"	1
	логическое "НЕ"	
7	Оператор «!» обозначает	
	логическое "И"	
	логическое "ИЛИ"	
	логическое "НЕ"	1

8	"не равно" в СИ обозначается	
	==	
	<>	
	><	
	!=	1
9	pow(3,3)%5=	
	4.	
	5.	
	2.	1
	3.	
10	Описание вещественного типа с СИ	
	float	1
	int	
	real	
	char	
11	Какой спецификатор нужно использовать для вывода вещественного числа?	
	%d	
	%f	1
	%c	
	%i	
12	Какой спецификатор нужно использовать для вывода целого числа?	
	%d	1
	%f	
	%c	

	%s	
13	Какой спецификатор нужно использовать для вывода одного символа?	
	%d	
	%f	
	%c	1
	%i	
14	Требуется вывести вещественное число в СИ, так чтобы после запятой было 3 знака. Какой формат нужно использовать?	
	%3d	
	%3f	
	%4.3f	1
	%4.3d	
15	В каком формате появится число на экране при использовании команды: printf("%4.2f",6);	
	6.	
	6.00	1
	6.00000	
	6.000000	
16	Какой оператор в СИ находит остаток от деления?	
	/	
	mod	
	%	1
	«&&»	
17	Описание строки s из 10 символов в СИ нужно выполнить:	

	string s(10);	
	char s[10];	
	char s[11];	1
	string s[10];	
18	\n в СИ обозначает:	
	горизонтальная табуляция	
	перевод курсора на новую строку	1
	спецификатор целочисленного формата	
	спецификатор вещественного формата	
19	Оператор соотношения "равно" записывается:	
	=	
	:=	
	!=	
	==	1
20	Функция strlen() ...	
	вычисляет длину строки	1
	выполняет сцепление строк	
	проверяет вхождение одной строки в другую	
	выполняет преобразование числа в строку	
21	Как записывается логическое "ИЛИ" в СИ?	
	ili	
	OR	
		1

22	Как записывается логическое "НЕ" в СИ?	
	!	1
	not	
	&&	
23	Язык СИ разработал...	
	Том Куртц	
	Никлаус Вирт	
	Деннис Ритчи	1
	Билл Гейтс	
24	15%3=...	
	0.	1
	1.	
	5.	
	3.	
25	Какую библиотеку нужно подключить для работы операторов scanf() и printf() в СИ?	
	<math.h>	
	<stdio.h>	1
	<conio.h>	
	<stdlib.h>	
26	Какую библиотеку нужно подключить для работы оператора fabs() в СИ?	
	<math.h>	1

	<stdio.h>	
	<conio.h>	
	<stdlib.h>	
27	Как находится модуль вещественного числа в СИ?	
	mod	
	fabs	1
	abs	
28	Как записать на СИ "корень квадратный из x"?	
	$x^{0.5}$	
	power(x,2)	
	pow(x,0.5)	1
	pow(x,y,2)	
29	Оператор форматированного вывода:	
	scanf()	
	print()	
	printf()	1
	write()	
30	Оператор ввода:	
	scanf()	1
	read()	
	printf()	
	input()	

31	Укажите правильно заданную команду для ввода вещественной переменной a.	
	<code>printf("%f",a);</code>	
	<code>scanf("%d",&a);</code>	
	<code>scanf("%f",&a);</code>	1
	<code>scanf("%-f",a);</code>	
32	Укажите правильно заданную команду для вывода вещественной переменной a.	
	<code>printf("%f",a);</code>	1
	<code>scanf("%d",&a);</code>	
	<code>scanf("%f",&a);</code>	
	<code>printf("%d",a);</code>	
33	Значение вещественной переменной нужно вывести следующим образом: выравнивание по левому краю, три знака после запятой. Выберите соответствующую команду.	
	<code>printf("%f5.3",x);</code>	
	<code>printf("%d3",x);</code>	
	<code>printf("%f-8.3",x);</code>	1
	<code>printf("%-3f",x);</code>	
34	Служебное слово switch используется в конструкции...	
	условия	
	множественного выбора	1
	цикла с параметром	
	форматированного вывода	
35	Выберите правильно записанное условие проверки числа x на четность на языке СИ:	

	if (x%2=0)	
	if (x%2==2)	
	if (x mod 2 =0)	
	if (x%2==0)	1
36	Переход на новую строку в языке Си осуществляется:	
	\t	
	%d	
	\n	1
	&&	
37	Укажите количество ошибок при использовании условного оператора на языке Си: <pre> if (a==5) { printf("Good"); b=a; } </pre>	
	1.	
	2.	
	3.	
	нет ошибок	1
38	Укажите неправильный вариант для обозначении имен переменных: A: VAZ B: _swww C: 5a D: re4 E: kl*s	
	С и E	1
	Только E	
	Все неправильные	
	Все правильные	

39	Каким словом обозначается «главная» функция в программе на языке Си:	
	<i>mail</i>	
	<i>main</i>	1
	<i>#include</i>	
	<i>return</i>	
40	Укажите какое значение будет выведено на экран: <pre>h=4.3767; j=0; printf("%1.1f",h);</pre>	
	4.0	
	0.3767	
	0.0	
	4.4	1
41	Укажите значение элемента массива K[2] после завершения цикла: <pre>int K[78]; for(i=0;i<=34;i+=2) K[i]=i-10;</pre>	
	-8	1
	2	
	34	
	-10	
42	Укажите количество ошибок в строке, подключающей модуль для работы операторов ввода и вывода: <code>#incude <studio.n></code>	
	1.	
	2.	
	3.	1
	нет ошибок.	

43	Целый тип в языке Си обозначается:	
	<i>integer</i>	
	<i>float</i>	
	<i>int</i>	1
	<i>Такого типа не существует</i>	
44	Укажите количество ошибок при использовании условного оператора на языке Си: <pre> if (a<-3); { printf("%d\n",a); a++; } </pre>	
	1.	1
	2.	
	3.	
	нет ошибок.	
45	Укажите количество ошибок при использовании оператора цикла на языке Си: <pre> for(t=-4,t<4;t=t+0.5 { printf("%f\n",t); } </pre>	
	1.	
	2.	1
	3.	
	4.	
46	Укажите неправильный вариант для обозначении имен переменных: A: a45 B: y_22 C: pyc D: f%	

	E: _1	
	B	
	C	
	D	
	C и D	1
47	Подключение какой библиотеки необходимо для использования математических функций:	
	<i>math.h</i>	1
	<i>stdio.h</i>	
	<i>#include</i>	
	<i>conio.h</i>	
48	Укажите какое значение будет выведено на экран: h=-6.278; j=6; printf(“%4.2f”,j);	
	6.00	1
	-6.28	
	-6.00	
	6.28	
49	Укажите значение элемента массива K[13] после завершения цикла: int K[13]; for(i=0;i<13;i=i++) K[i]=i;	
	13.	
	0.	
	Массив задан неправильно	

	Элемент К[13] не определится	1
50	Укажите количество ошибок в строке: print("%d %d",y,&y);	
	1.	
	2.	1
	3.	
	Ошибок нет.	
51	Для подключения модулей используется:	
	#include	1
	stdio.h	
	main	
	getch()	
52	Тип переменной указывается:	
	через : после переменной	
	после переменной	
	указывать не обязательно	
	до ее написания	1
53	Укажите количество ошибок при использовании условного оператора на языке Си: <pre> if (c<>3) { G=sin(c); printf("%f\n",G); c--; }; </pre>	
	1.	
	2.	1

	3.	
	Ошибок нет.	
54	Укажите количество ошибок при использовании оператора цикла на языке Си: <pre> for (g=0,g<13,g++); { printf("%f\n",g); } </pre>	
	1.	
	2.	
	3.	1
	4.	
55	Подключение какой библиотеки необходимо для использования функции getch()?	
	<i>stdio.h</i>	
	<i>stdlib.h</i>	
	<i>getch.h</i>	
	<i>conio.h</i>	1
56	Укажите значение элемента массива K[0] после завершения цикла: <pre> int K[4]; for(i=3;i>0;i=i-1) K[i]=i*i; K[0]=K[2]; </pre>	
	4.	1
	2.	
	0.	
	1.	
57	Укажите количество ошибок при использовании условного оператора: <pre> if e=12 printf("%d",e); </pre>	

	1.	
	2.	1
	4.	
	Нет ошибок.	
58	Для использования функции strlen() необходимо подключить библиотеку:	
	stdio.h	
	conio.h	
	string.h	1
	stdlib.h	
59	Укажите диапазон случайных чисел для random(12)-6:	
	от 0 до 6	
	от -6 до 0	
	от -6 до 5	1
	от -6 до -1	
60	Укажите количество ошибок при использовании условного оператора на языке Си: <pre> while (d!=13) { G=cos(sqrt(d)); printf("%f\n",G); d--; } </pre>	
	1.	
	2.	
	3.	
	Нет условного оператора.	1

61	<pre>while (d!=13) { G=cos(sqrt(d)); printf(“%f\n”,G); d--; }</pre> Какой оператор используется в коде программы?	
	оператор условия	
	оператор цикла с параметром	
	оператор цикла с предусловием	1
	оператор цикла с постусловием	
62	Укажите значение элемента массива K[4] после завершения цикла: <pre>int K[100]; for(i=3;i<100;i++) { if(i==4) K[i]=10; else K[i]=2*i; }</pre>	
	100.	
	4.	
	8.	
	10.	1
63	Каким символом завершается программа на языке Си:	
	«}»	1
	«getch»	
	«.»	
	«return»	
64	Можно ли в операторе условия if не ставить скобки {}	
	Всегда надо ставить	
	Можно, если нет else	

	Нельзя	
	Можно, если блок состоит только из одного оператора	1
65	Целочисленный массив обозначается:	
	<code>float A[4]</code>	
	<code>int A[13]</code>	
	<code>int A[1][3]</code>	
	2 или 3 вариант	1
66	Укажите количество ошибок при использовании условного оператора на языке Си: <pre> if (x<10) { printf("%f\n",x); x++; } else printf("end"); </pre>	
	1.	
	2.	
	3.	
	Нет ошибок.	1
67	Какой модуль необходимо подключить для функции random?	
	<code>stdio.h</code>	
	<code>stdlib.h</code>	1
	<code>conio.h</code>	
	<code>randomize.h</code>	
68	Какой из операторов не является оператором цикла?	
	<code>for</code>	
	<code>while</code>	

	<i>if</i>	1
	<i>2 и 3 варианты</i>	
69	Двумерный массив обозначается:	
	W[3][5]	1
	W[3;5]	
	W[3,5]	
	W[3],[5]	
70	В каком порядке следует подключать модули?	
	<i>В порядке следования необходимых функций</i>	
	<i>В строго определённом</i>	
	<i>Не имеет значения</i>	1
	<i>Их можно вообще не подключать</i>	
71	a=5 b=7 После выполнения операции a+=b значение переменной a равно:	
	12.	1
	5.	
	35.	
	операция не имеет смысла	
72	<pre>int a,b; float c; printf("Vvedite a\n"); scanf("%d",&a); printf("Vvedite b\n"); scanf("%d",&b); c=a/b; printf("c=%3.2f\n",c);</pre> С клавиатуры вводятся 12 и 5. Какой результат выведется на экран?	
	c=0.05	

	c=2.00	1
	c=2.40	
	c=3.20	
73	В каком выражении использован оператор инкремента?	
	4*a--	
	a*=b	
	x*y++	1
	ceil(x)	
74	Оператор, уменьшающий значение переменннной на единицу - это ...	
	fabs	
	инкремента	
	floor	
	дкремента	1

Пакет преподавателя:

№ вопроса	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	26	27	28	29
Верный ответ	1	3	1	2	1	2	3	4	3	1	2	1	3	3	3	2	3	3	2	4	1	4	2	3	1	2	1	2	3

№ вопроса	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58
Верный ответ	3	1	3	1	3	2	4	3	4	1	2	4	1	3	3	1	2	4	1	1	4	2	1	4	2	3	4	1	2

№ вопроса	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74
Верный ответ	3	3	4	3	4	1	4	4	4	2	1	3	1	2	3	4

ОЦЕНОЧНОЕ СРЕДСТВО № 18

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ СОЗДАНИЯ И ЗАЩИТЫ ПРЕЗЕНТАЦИИ

Раздел	Раздел 3 Программирование на языке С Основные модели алгоритмов
Тема	Тема 3.6. Программирование с использованием структур
Форма контроля	Исследовательское групповое задание
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ОК	ОК 6,7,8
2. Знания	4

Условия выполнения задания

задание выполняется вне аудитории, 4 часа

Инструкция для студентов

1. Последовательность и условия выполнения задания

студенты создают презентацию в группе по 3 человека, затем защищают ее. Защита презентации на теоретическом занятии.

2. Вы можете воспользоваться: *Конспект лекции, Интернет*

3. Максимальное время выполнения задания: 4 часа.

Оборудование и оснащение: *ПК, Интернет*

ОЦЕНОЧНОЕ СРЕДСТВО № 19

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ПРАКТИЧЕСКОГО ЗАДАНИЯ

Раздел	Раздел 1 Алгоритм. Исполнители алгоритма.
Тема	Тема 1.1. Алгоритм. Базовые алгоритмические конструкции
Форма контроля	Практическое задание
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7. ПК 2.8
2. Знания	1
4. Умения	1,2

Условия выполнения задания

Аудитория, 2 часа.

Инструкция для студентов

1. Последовательность и условия выполнения задания: *изучение и выполнение заданий, составление отчета, студенты, выполнившие и сдавшие практическое задание раньше остальных, могут получить дополнительную оценку, выполнив задания из пункта «Дополнительно на оценку»*
2. *Вы можете воспользоваться конспектами, учебником*
3. Максимальное время выполнения задания: *2 часа.*
4. Перечень раздаточных и дополнительных материалов *методические указания к лабораторным работам*

Оборудование и оснащение: *ПК*

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

Цель работы: Научиться составлять циклические алгоритмы, создавать программы, используя полученные знания.

Содержание работы.

Основные понятия.

1 Графический способ представления циклических алгоритмов (метод блок-схемы).

Проверка условий	
Цикл с параметром	

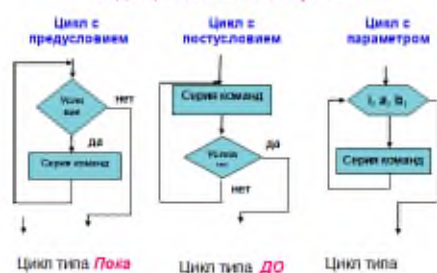
2 Алгоритм, в котором вычисления повторяются по одной и той же совокупности формул, называется циклическим

3 Существуют следующие конструкции для организации циклов:

- цикл с предусловием;
- цикл с постусловием;
- цикл со счетчиком.

4 Графическое представление циклических алгоритмов

Виды циклических алгоритмов



5 Циклы с предусловием используются тогда, когда выполнение цикла связано с некоторым логическим условием. Оператор цикла с предусловием имеет две части: условие выполнения цикла и тело цикла. При выполнении оператора цикла определенная группа операторов выполняется до тех пор, пока

Практическое задание

определенное в операторе условие истинно. Если условие сразу ложно, то оператор не выполнится ни разу.

Составить и записать алгоритм решения задачи в графическом и словесно-формульном виде

Исходные данные:

Дано целое число N . Вычислить значение выражения $1 + \frac{2}{2!} + \frac{3}{3!} + \dots + \frac{N}{N!}$, результат вывести как действительное число.

Решение:

$$1 \quad N! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot N$$

Значение выражения будем рассчитывать поэтапно.

$$\text{На первом шаге к значению суммы } S \text{ добавим } 1 = \frac{1}{1!} = \frac{1}{1}$$

$$\text{На втором } \frac{2}{2!} = \frac{2}{1 \cdot 2}$$

$$\text{На третьем } \frac{3}{3!} = \frac{3}{1 \cdot 2 \cdot 3}$$

...

$$\text{На } N\text{-ном } \frac{N}{N!} = \frac{N}{1 \cdot 2 \cdot \dots \cdot N}$$

В зависимости от заданного пользователем значения N количество шагов в алгоритме может быть разным, поэтому для определения шага добавим промежуточную переменную I , которая будет меняться от 1 до N

Так как к значению суммы на каждом шаге мы добавляем какое-то число, необходимо определить начальное значение, не влияющее на результат. Ясно, что это $S = 0$.

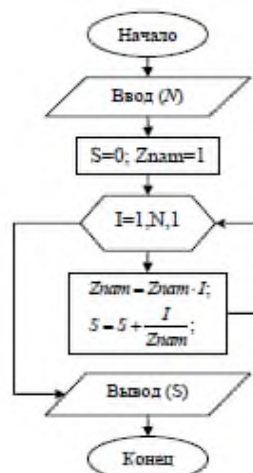
На каждом шаге считать факториал трудно, поэтому обратим внимание, что на любом этапе вычислений значение факториала отличается от предыдущего на число, равное значению шага, значит введем переменную для факториала: $Z_{\text{nam}} = 1$.

Получили общие формулы:

$$Z_{\text{nam}} = Z_{\text{nam}} \cdot I;$$

$$S = S + \frac{I}{Z_{\text{nam}}};$$

Задание



1. Начало;
2. Ввод (N);
3. $S=0$;
4. $Z_{\text{nam}}=1$;
5. Для $I = 1$ до N выполнить
нц
 $Z_{\text{nam}} = Z_{\text{nam}} \cdot I$;
 $S = S + \frac{I}{Z_{\text{nam}}}$;
 кц
6. Вывод (S);
7. Конец.

Дополнительно на оценку

Задания к практической работе.

- 1 Даны действительные числа x , y . Вывести в порядке возрастания все целые числа, расположенные между x и y , а также количество этих чисел.
- 2 Даны действительные числа x , y . Вывести в порядке убывания все целые числа, расположенные между x и y , а также количество этих чисел.
- 3 Дано действительное число – цена 1 кг конфет. Вывести стоимость 1, 2 ... 15 кг конфет.

ОЦЕНОЧНОЕ СРЕДСТВО № 20

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ПРАКТИЧЕСКОГО ЗАДАНИЯ

Раздел	Раздел 2 Программирование на языке Pascal.
Тема	Тема 1.1. Алгоритм. Базовые алгоритмические конструкции
Форма контроля	Практическое задание
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7. ПК 2.8
2. Знания	3
4. Умения	1,2

Условия выполнения задания

Аудитория, 2 часа.

Инструкция для студентов

1. Последовательность и условия выполнения задания: изучение и выполнение заданий, составление отчета, студенты, выполнившие и сдавшие практическое задание раньше остальных, могут получить дополнительную оценку, выполнив задания из пункта «Дополнительно на оценку»
2. Вы можете воспользоваться конспектами, учебником

3. Максимальное время выполнения задания: 2 часа.

4 Перечень раздаточных и дополнительных материалов *методические указания к лабораторным работам*

Оборудование и оснащение: ПК

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

Практическое задание

Массив - это структура данных, что представляет собой совокупность фиксированного размера и конфигурации упорядоченных однородных независимых переменных.

Массив относится к так называемым структурированным данным, то есть таких, что имеют фиксированную внутреннюю структуру (организацию).

Массив характеризуется:

1. Количеством размерностей (количеством координат, необходимых для определения местонахождения нужного элемента массива).
2. Общим идентификатором (именем) для всех элементов массива.
3. Индексом или совокупностью индексов, которые определяют каждый отдельный элемент массива.

Одномерный массив (вектор) - имеет одну размерность

При обращении к отдельному элементу массива необходимо указать его индекс (местонахождение в массиве):

$A[7] \quad i:=7; A[i]$

Здесь i - индекс элемента массива, может быть только целого или натурального типа

Двумерные массивы (матричные) - имеют две размерности, $m \times n$. Доступ к отдельному элементу массива осуществляется путем определения двух его координат: номера строки $i=1..m$ и столбца $j=1..n$

Операции предоставления выполняются аналогично:

$a[3,9]=8; B:=A[1,1]$

В трехмерном массиве для доступа к элементу необходимо указать три индекса $A[i,j,k]$. Можно создавать массивы с большей размерностью, но работа с массивами, размерность которых превышает 3, существенным образом усложняет алгоритм, поэтому, по возможности, необходимо избегать организации подобных структур данных.

Сортировка массивов

Методы сортировки можно разбить в соответствии с определяющими их принципами на три основные группы:

1. Сортировка с помощью вставки (by Insertion) или с помощью включения
2. Сортировка с помощью выбора (by Selection) или с помощью выделения
3. Сортировка с помощью обмена (by Exchange) или пузырьковая.

Каждая группа имеет прямой метод (самый простой) и улучшенный(усложненный) методы сортировки

I. Сортировка с помощью вставки

Принцип сортировки: массив распределяется на отсортированную и неотсортированную части. На первом шаге за отсортированную часть (последовательность) принимается первый элемент массива. Каждый следующий элемент из неотсортированной части вставляем в заранее отсортированную последовательность так, чтобы эта последовательность оставалась отсортированной.

При этом надо:

1. Найти место, куда нужно вставить этот элемент
2. Сдвинуть элементы, которые стоят справа в отсортированной части на одну позицию вправо.
3. На освобожденное место поставить элемент, который анализируется (вставляется).

Два способа выполнения этих действий:

1) каждый следующий элемент сравнивается с элементами в отсортированной части, находится место вставки, все следующие элементы сдвигаются на одну позицию вправо и после этого вставляется элемент;

2) элемент, который вставляется, последовательно, слева направо, сравнивается с любым из элементов в отсортированной части. Если нужно, элемент в отсортированной части сразу сдвигается на одну позицию вправо. Как только найдено нужное место вставки, элемент, который анализируется, вставляется на нужную позицию.

II. Сортировка с помощью прямого выбора

Принцип сортировки: массив также делится на отсортированную и неотсортированную части. На первом шаге весь массив - неотсортированный. В неотсортированной части находится минимальный (или максимальный) элемент и меняется местами с первым элементом неотсортированной части. Граница отсортированной части сдвигается вправо на 1. Процедура выполняется циклически, n-1 раз (последний элемент передвигать не надо)

III. Сортировка с помощью прямого обмена (пузырьковая)

Принцип сортировки: слева направо поочередно направляется сравнение двух соседних элементов. Если они не упорядочены между собой, то меняются местами. В базовом алгоритме прохождения массива и очередное приведение в порядок повторяются n-1 раз.

ПРИМЕР ПРОГРАММЫ

2.Нахождение среднего арифметического ста чисел

Program N1;

var M: Array [1..100] of Integer; A: Real; I: Byte;

begin Randomize;

For I := 1 to 100 do M[I] := Random(500);

For I := 1 to 100 do

A := A + M[I];

A := A / 100;

Write('Среднее арифметическое всех чисел массива: ', A);

end.

3. Найдите максимальное число среди введенных чисел и выведите его на экран.
Program N2;

```
var M: Array [1..10] of Integer; Max: Integer; I: Byte;
```

```
begin Writeln('Введите 10 чисел: ');
```

```
For I := 1 to 10 do
```

```
begin Write('N',i,': ');
```

```
Readln(M[i]);
```

```
end;
```

```
Max := M[1];
```

```
For I := 1 to 10 do
```

```
if Max < M[i] then Max := M[i];
```

```
Write('Максимальное число: ', Max);
```

```
Readln;
```

```
end
```

Задание №1

Найдите среднее арифметическое первых двухсот чисел.

Дополнительные задания на оценку

1.Задание № 2

Найдите минимальное число среди введенных 10 чисел.

2.Задание № 3

Найдите максимальное число среди введенных 15 чисел.

ОЦЕНОЧНОЕ СРЕДСТВО № 21

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ПРАКТИЧЕСКОГО ЗАДАНИЯ

Раздел	Раздел 2 Язык программирования Pascal
Тема	Тема 2.11. Программирование графики
Форма контроля	Практическое задание
Вид контроля	оперативный
Объекты оценки:	
1. Компоненты ПК	ПК 2.7, ПК 2.8
2. Знания	1
4. Умения	1,2

Условия выполнения задания

Аудитория, 2 часа.

Инструкция для студентов

1. Последовательность и условия выполнения задания: *изучение и выполнение заданий, составление отчета, студенты, выполнившие и сдавшие практическое задание раньше остальных, могут получить дополнительную оценку, выполнив задания из пункта «Дополнительно на оценку»*
2. *Вы можете воспользоваться конспектами, учебником*
3. Максимальное время выполнения задания: *2 часа.*
- 4 *Перечень раздаточных и дополнительных материалов методические указания к лабораторным работам*

Оборудование и оснащение: *ПК*

Источники

Семакин И.Г., Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования – М.: Издательский центр «Академия», 2014 – 304с.

Практическое задание

Для формирования графических изображений в Турбо Паскале предназначен стандартный модуль GRAPH. В нем содержится 79 графических процедур, функций, десятки стандартных констант и типов данных. В состав модуля GRAPH входит ряд программ драйверов для наиболее распространенных видеоадаптеров: CGA, MCGA, EGA, VGA, HERCULES, AT&T, 3270 PC. Эти драйверы хранятся в файлах с расширением BGI.

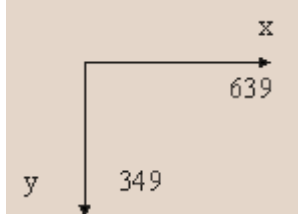
Скелет графической программы:

```
program имя;  
USES GRAPH; {подключение модуля}  
var GR,GM:integer;  
begin  
GR:=detect;  
initgraph(gr,gm,' '); {вызов процедуры GRAPH}  
графические операторы  
readln;  
closegraph;  
end.
```

INITGRAPH(драйвер, режим, 'путь') - подключает графический режим. Драйвер - это переменная целого типа, определяющая тип графического дисплея. Режим - это переменная типа integer, определяющая режим работы графического адаптера(GM). Путь - указывает путь программ с TPU.

DETECT - режим автоматического определения типа.

CLOSEGRAPH - прекращает работу графического адаптера и восстанавливает текстовый режим работы экрана.



Цвет

0 - черный	11 - светло-голубой
1 - синий	12 - светло-красный
2 - зеленый	13 - светло-лиловый
3 - голубой	14 - желтый
4 - красный	15 - белый
5 - лиловый	
6 - коричневый	
7 - светло-серый	
8 - темно-серый	
9 - светло-синий	
10 - светло-зеленый	

ГРАФИЧЕСКИЕ ОПЕРАТОРЫ:

1. SETCOLOR(цвет) - устанавливает текущий цвет для выводимых линий и символов;
2. SETBKCOLOR(цвет) - устанавливает цвет фона;
3. MOVETO(x,y) - переводит курсор в позицию x,y, точка при этом не высвечивается;
4. LINETO(x,y) - рисует линию от текущей позиции курсора до x,y;
5. LINEREL(Dx,Dy) - рисует линию от текущей позиции на относительное расстояние Dx,Dy;
6. PUTPIXEL(x,y,цвет) - выводит точку;
7. LINE(x1,y1,x2,y2) - выводит линию;
8. CIRCLE(x,y,r) - выводит окружность;
9. ARC(x,y,начальный угол,конечный угол,r) - выводит дугу (углы измеряются в градусах против часовой стрелки);
10. ELLIPSE(x,y,нач.угол,кон.угол,Rx,Ry) - эллипс;
11. RECTANGLE(x1,y1,x2,y2) - прямоугольник, где x1,y1 - координаты левого верхнего угла, x2,y2 - правого нижнего угла;
12. BAR(x1,y1,x2,y2) - закрашенный прямоугольник;
13. BAR3D(x1,y1,x2,y2,глубина,false или true) - параллелепипед; где x1,y1 - координаты левого верхнего угла; x2,y2 - координаты правого нижнего угла, true - верхняя грань есть, false - верхняя грань отсутствует.
14. FILLELLIPSE(x,y,Rx,Ry) - закрашенный эллипс;
15. SECTOR(x,y,нач.угол,кон.угол,Rx,Ry) - сектор закрашенный;
16. SETFILLESTYLE(штриховка,цвет) - каким цветом или какими штрихами будет закрашена замкнутая область;

Штриховка:

- 0 - фоном
- 1 - сплошная
- 2 - _ _ _
- 3 - ///
- 4 - толстыми///
- 5 - \\\
- 6 - толстыми \\\
- 7 - ++++
- 8 - x x x
- 9 - квадратами
- 10 - редкими точками
- 11 - частыми точками
- 12 - пользовательские FLOODFILL(x,y,цвет,цвет границы) - штрихует произвольную замкнутую область;

Пример графических программ:Программа №1.

```
program g1;  
uses graph;  
var gr,gm:integer;  
begin  
gr:=detect;  
initgraph(gr,gm,' ');  
setcolor(3);  
setbkcolor(14);  
setfillstyle(8,4);  
fillellipse(100,100,30,50);
```

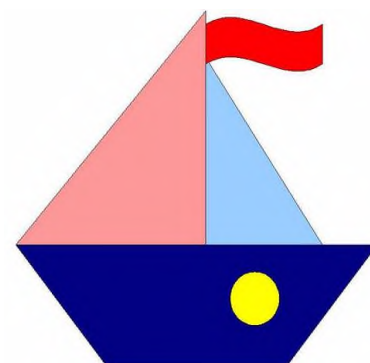
```

setfillstyle(2,2);
circle(200,200,50);
floodfill(200,200,3);
readln;
closegraph;
end.

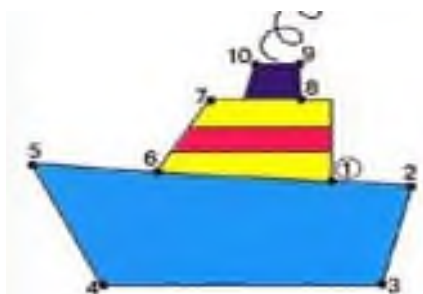
```

Задание

Нарисовать методами программирования:



Дополнительно на оценку



3. Контрольно-оценочные средства для промежуточной аттестации

Перечень вопросов для промежуточной аттестации

ОЦЕНОЧНОЕ СРЕДСТВО № 1

ПЕРЕЧЕНЬ ТЕОРЕТИЧЕСКИХ ВОПРОСОВ К ЭКЗАМЕНУ

- 1.Алгоритм. Базовые алгоритмические конструкции.
- 2.Исполнители алгоритма. СКИ. Пошаговое выполнение алгоритма.
- 3.История и классификация языков программирования.
- 4.Pascal. Алфавит. Идентификаторы, переменные, константы. Общая структура программы. Комментарии.
- 5.Pascal. Стандартные типы данных. Арифметические и логические выражения. Стандартные функции.
- 6.Pascal. Простые операторы. Операторы присваивания, составной и простой операторы, операторы ввода-вывода.
- 7.Pascal. Условный оператор.
- 8.Pascal. Оператор цикла с предусловием.
- 9.Pascal. Оператор цикла с постусловием.
- 10.Pascal. Оператор цикла с параметром.
- 11.Pascal. Оператор множественного выбора.
- 12.Pascal. Строки. Действия над строками.
- 13.Pascal. Множества. Операции над множествами.
- 14.Pascal. Одномерные массивы.
- 15.Pascal. Двумерные массивы. Свойства квадратных матриц.
- 16.Pascal. Записи. Оператор присоединения.
- 17.Pascal. Файлы. Функции для работы с файлами.
- 18.Pascal. Процедуры и функции.
- 19.Pascal. Работа с графикой.
- 20.Pascal. Анимация.
- 21.C. Основные типы данных. Стандартные функции. Математические функции. Структура программы.
- 22.C. Операторы форматного ввода и вывода. Спецификаторы и модификаторы формата.
- 23.C. Оператор присваивания. Арифметические выражения. Операторы инкремента и декремента. Сокращенный вид записи арифметических выражений.
- 24.C. Условные выражения. Оператор условия.
- 25.C. Оператор множественного выбора.
- 26.C. Циклы с предусловием.
- 27.C. Циклы с постусловием.
- 28.C. Циклы с параметром.
- 29.C. Массивы.
- 30.C. Строки.
- 31.C. Функции.
- 32.C. Структуры.
- 33.C. Файлы.

ОЦЕНОЧНОЕ СРЕДСТВО № 2

ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ ВОПРОСОВ К ЭКЗАМЕНУ

- 1) Составить программу для вычисления значения выражения:

$$y = \cos^2 2x + \left| \sin \frac{x}{2} \right|$$

Входные данные – x (вводится с клавиатуры). Выходные данные – y .

- 2) Составьте блок-схему алгоритма и программу, определяющую является ли введенное число двузначным. Результатом работы программы должно быть сообщение 'является' или 'не является'.

- 3) Составьте блок-схему и программу вычисления значения функции, используя алгоритм полного ветвления:

$$y = \begin{cases} \frac{9x^2 + 5}{3x + 12}, & \text{если } x < -4, \\ 4x^2 - 7, & \text{если } x \geq -4. \end{cases}$$

- 4) Составьте блок-схему алгоритма и программу вычисления площади и периметра прямоугольника, если известны его стороны (вводятся с клавиатуры).

- 5) Составьте блок-схему алгоритма и программу вычисления площади и периметра треугольника, если известны его стороны и высота (вводятся с клавиатуры).

- 6) Составьте блок-схему алгоритма и программу, определяющую является ли введенное целое число положительным. Результатом работы программы должно быть сообщение 'является' или 'не является'.

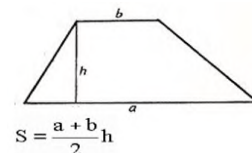
- 7) Составьте блок-схему алгоритма и программу, определяющую является ли введенное целое число нечетным. Результатом работы программы должно быть сообщение 'является' или 'не является'.

- 8) Составьте блок-схему алгоритма и программу вычисления количества букв в строке. Строка вводится с клавиатуры.

- 9) Составьте блок-схему алгоритма и программу вычисления площади ромба по его диагоналям (вводятся с клавиатуры).

$$S = \frac{1}{2} d_1 \cdot d_2$$

- 10) Составьте блок-схему алгоритма и программу вычисления площади трапеции по ее основаниям и высоте (вводятся с клавиатуры).



- 11) Составить блок-схему алгоритма и программу для вычисления значения выражения:

$$a = \sqrt{\frac{b+c}{5bc}} - \frac{b^2}{2c};$$

Входные данные – c, b (вводятся с клавиатуры). Выходные данные – a .

12) Составьте блок-схему алгоритма и программу, которая по введенному с клавиатуры целому числу в диапазоне 0 – 9 выводит строку — название соответствующей цифры на русском языке (0 — "ноль", 1 — "один", 2 — "два", ...).

13) Составьте блок-схему алгоритма и программу вычисления среднего арифметического трех целых чисел, введенных с клавиатуры, с точностью до трех знаков после запятой.

14) Составить блок-схему алгоритма и программу для вычисления значения выражения:

$$s = \frac{x + \sqrt{y}}{x^2 + y^2}$$

Входные данные – x, y (вводятся с клавиатуры). Выходные данные – s.

15) Составить блок-схему алгоритма и программу, которая по введенному номеру дня недели (вводится с клавиатуры), выдает его название.

16) Составьте блок-схему алгоритма и программу вычисления площади круга и длины окружности по введенному радиусу (вводится с клавиатуры).

17) Составьте блок-схему алгоритма и программу вычисления площади и периметра квадрата, если известны его стороны (вводятся с клавиатуры).

18) Составьте блок-схему алгоритма и программу вывода на экран всех чисел, которые кратны числу 5, в интервале от a до b (значения a и b вводятся, a < b).

19) Составьте блок-схему алгоритма и программу вычисления произведения целых чисел из промежутка [-6; 5).

20) Составьте блок-схему алгоритма и программу вывода на экран всех трехзначных чисел, кратных трем. Решите данную задачу, используя цикл с параметром, цикл с предусловием и цикл с постусловием.

21) Составьте блок-схему алгоритма и программу расположения трех чисел в порядке возрастания.

22) Составьте блок-схему алгоритма и программу вывода на экран таблицы степеней 2^n , где $0 \leq n \leq 10$.

23) Составьте блок-схему алгоритма и программу вывода таблицы значений функции $y = 4x^2 + 5x - 10$ на отрезке [-9; 9] с шагом $h = 3$. Вывод результатов оформите в виде таблицы с границами.

24) Составьте блок-схему алгоритма и программу вычисления среднего арифметического целых чисел из отрезка [-4; 15] с точностью до трех знаков после запятой.

25) Составьте блок-схему алгоритма и программу вывода таблицы значений функции $y = x^2 - 5x - 2$ на отрезке [1; 20] с шагом $h = 2$. Вывод результатов оформите в виде таблицы с границами.

26) Составьте блок-схему алгоритма и программу вычисления значения выражения для данного натурального числа N: $1/1^2 + 1/2^2 + 1/3^2 + \dots + 1/N^2$. Значение N вводится с клавиатуры.

27) Составьте блок-схему алгоритма и программу заполнения массива, не используя клавиатуру, числами: 2, 4, 6, 8, 10, 12, 14, 16, 18, 20. Найти сумму элементов полученного массива.

28) Составьте блок-схему алгоритма и программу, определяющую, на какую букву начинается второе слово в строке, введенной с клавиатуры.

29) Составить программу и блок-схему алгоритма. Сгенерировать массив A из 10 целых чисел, которые берутся из промежутка [-15, 15]. Подсчитать количество нечетных элементов массива.

30) Составить программу и блок-схему алгоритма. Сгенерировать массив C из 15 целых чисел, которые берутся из промежутка [-10, 10]. Подсчитать сумму четных элементов массива.

Рецензия

на рабочую программу по учебной дисциплине

ОП.04 Основы алгоритмизации и программирования

для специальности 09.02.07 Информационные системы и программирование

Разработчик: Кожевин В.Г., преподаватель ГБПОУ НАО «Ненецкий аграрно-экономический техникум им. В.Г. Волкова»

Рабочая программа учебной дисциплины ОП.04 Основы алгоритмизации и программирования разработана на основе примерной рабочей программы дисциплины «Основы алгоритмизации и программирования» для профессиональных образовательных организаций, одобренной ФГБОУ ДПО ИРПО от 29 сентября 2022 по специальности 09.02.07 Информационные системы и программирование, входящей в состав укрупнённой группы специальностей 09.00.00 Информатика и вычислительная техника, а также в соответствии с базисным учебным планом по специальности и рабочим учебным планом ГБПОУ НАО «Ненецкий аграрно-экономический техникум имени В.Г. Волкова» по специальности.

Структура рабочей программы отвечает требованиям, предъявляемым нормативным документом Министерства образования и науки РФ «Разъяснения по формированию примерных программ учебных дисциплин на основе ФГОС НПО и СПО» к составлению программ учебных дисциплин и состоит из следующих разделов: титульный лист, паспорт рабочей программы учебной дисциплины, структуру и содержание учебной дисциплины, условия реализации учебной дисциплины, контроль и оценка результатов освоения учебной дисциплины.

Тематически учебный материал сгруппирован по разделам: «Введение в программирование», «Основные конструкции языков программирования», «Структурное и модульное программирование». Структурирование содержания учебного материала в программе дидактически целесообразно и логично. Основной акцент сделан на формирование и развитие ОК и ПК.

Данная рабочая программа может быть использована для изучения дисциплины ОП.04 Основы алгоритмизации и программирования по специальности 09.02.07 Информационные системы и программирование.

Рецензент: преподаватель ГБПОУ НАО
«Нарьян-Марский социально-гуманитарный
колледж имени И.П. Выучейского»

/Панова Н.А./

ЛИСТ ЭКСПЕРТНОЙ ОЦЕНКИ
РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ
(СОДЕРЖАТЕЛЬНАЯ ЭКСПЕРТИЗА)

Проведена экспертная оценка рабочей программы учебной дисциплины ОП.04. Основы алгоритмизации и программирования, специальности 09.02.07 Информационные системы и программирование

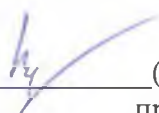
Разработчик: Кожевин Владимир Григорьевич

Образовательное учреждение: ГБПОУ НАО «Ненецкий аграрно-экономический техникум имени В.Г. Волкова»

Критерии оценки РПУД		да	нет	отсутствует	Примечания
Экспертиза титульного листа					
-	наименование учредителя ОУ указано верно	+			
-	наименование учредителя ОУ оформлено в соответствии с ГОСТ ОРД	+			
-	наименование ОУ соответствует уставу ОУ	+			
-	реквизиты лицевой и обратной сторон титульного листа рабочей программы оформлены в соответствии с требованиями ГОСТ ОРД	+			
-	наименование учебной дисциплины совпадает с наименованием во ФГОС	+			
-	наименование дисциплины вариативной части совпадает с наименованием в рабочем учебном плане и не противоречит требованиям ФГОС			+	Не предусмотрено
	обратная сторона титульного листа содержит:				
-	перечень документов, на основании которых разработана рабочая программа учебной дисциплины	+			
-	наименование организации-разработчика рабочей программы учебной дисциплины в соответствии с уставом ОУ	+			
-	фамилию, имя и отчество разработчика программы (одного или нескольких), ученую степень, звание, должность	+			
Экспертиза Раздела 1. Паспорт программы учебной дисциплины					
-	Раздел 1. «Паспорт программы учебной дисциплины» содержит все пункты и оформлен в соответствии с форматом разъяснений МОН РФ	+			
-	Перечень профессий/специальностей в пункте 1.1. «Область применения программы» представлен полно и достоверно, в соответствии с перечнем профессий/специальностей НПО/СПО	+			
-	В пункте 1.1. указаны возможности использования программы в профессиональном образовании и/или обучении	+			
-	Пункт 1.2. «Место учебной дисциплины в структуре основной профессиональной образовательной программы» указывает на принадлежность дисциплины к учебному циклу.	+			
-	Пункт 1.3. «Цели и задачи учебной дисциплины – требования к результатам освоения учебной дисциплины» соответствует требованиям ФГОС	+			
-	Перечень умений и знаний в вариативной части конкретизирует и/или расширяет требования ФГОС (при наличии вариативной части)			+	Не предусмотрено
-	Наименование и содержание дисциплины вариативной части не совпадает с инвариантной частью			+	Не предусмотрено
-	Пункт 1.4. «Количество часов на освоение рабочей программы учебной дисциплины» содержит распределение часов на обязательную аудиторную нагрузку, самостоятельную работу.	+			

	Экспертиза Раздела 2. Структура и содержание учебной дисциплины				
-	Таблица 2.1. «Объем учебной дисциплины и виды учебной работы» содержит почасовое распределение видов учебных работ в соответствии с разъяснениями МОН РФ	+			
-	Структура содержания учебной дисциплины не противоречит принципу практикоориентированности обучения (БЖД в СПО 64/48 – рекомендации по разработке БУП СПО п. 1.2)	+			
-	Таблица 2.2. «Тематический план и содержание учебной дисциплины» отражает содержание учебной дисциплины	+			
-	Содержание учебной дисциплины соответствует требованиям к умениям и знаниям ФГОС	+			
-	Содержание учебной дисциплины вариативной части не противоречит подготовке по профессии/специальности			+	
-	Структурирование содержания учебного материала в программе дидактически целесообразно и логично	+			
-	Тематика курсовых работ соответствует целям и задачам освоения учебной дисциплины (пункт заполняется, если в программе дисциплины предусмотрена курсовая работа, для СПО).			+	Не предусмотрено
-	Уровни усвоения всех дидактических единиц проставлены	+			
	Экспертиза раздела 3 «Условия реализации учебной дисциплины»				
-	Раздел 3 «Условия реализации учебной дисциплины» содержит все пункты в соответствии с разъяснениями МОН РФ	+			
-	Пункт 3.1. «Требования к минимальному материально-техническому обеспечению» соответствует содержанию учебной дисциплины	+			
-	Перечисленное оборудование и средства обучения кабинетов и лабораторий обеспечивают проведение всех видов занятий, предусмотренных программой учебной дисциплины.	+			
-	Пункт 3.2. «Информационное обеспечение обучения» содержит перечень учебных изданий, Интернет-ресурсов, дополнительной литературы	+			
-	Основные и дополнительные источники соответствуют содержанию программы учебной дисциплины	+			
-	Основные и дополнительные источники оформлены в соответствии с требованиями стандартов.	+			
	Экспертиза раздела 4 «Контроль и оценка результатов освоения дисциплины»				
-	Результаты обучения (освоенные умения, усвоенные знания) представлены в полном объеме (перечислены все знания и умения, указанные в паспорте программы)	+			
-	Комплекс форм и методов контроля и оценки освоенных умений и усвоенных знаний соответствует объектам оценки.	+			
	Заключение эксперта: Рекомендовано к использованию	+			

Замечания и рекомендации:

Эксперт:  (Н.А. Панова),

преподаватель ГБПОУ НАО «Нарьян-Марский социально-гуманитарный колледж

имени И.П. Выучейского»

Дата «24» мая 2024 г.

**ЛИСТ ЭКСПЕРТНОЙ ОЦЕНКИ
РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ
(ТЕХНИЧЕСКАЯ ЭКСПЕРТИЗА)**

Проведена экспертная оценка рабочей программы учебной дисциплины ОП.04. Основы алгоритмизации и программирования, специальности 09.02.07 Информационные системы и программирование Разработчик: Кожевин Владимир Григорьевич
Образовательное учреждение: ГБПОУ НАО «Ненецкий аграрно-экономический техникум имени В.Г. Волкова»

Критерии оценки РПУД	Экспертная оценка	
	да	нет
Экспертиза оформления титульного листа и раздела «Содержание»		
Титульный лист представлен	+	
Наименование программы учебной дисциплины на титульном листе совпадает с наименованием дисциплины в тексте ФГОС	+	
Оборотная сторона титульного листа представлена и оформлена	+	
На лицевой и оборотной стороне титульного листа реквизиты представлены	+	
Нумерация страниц в «Содержании» соответствует размещению разделов программы	+	
Экспертиза Раздела 1. Паспорт рабочей программы учебной дисциплины		
Раздел 1. «Паспорт рабочей программы учебной дисциплины» представлен	+	
Пункт 1.1. «Область применения программы» представлен.	+	
Возможности использования программы представлены	+	
Перечень профессий / специальностей в пункте 1.1. «Область применения программы» представлен	+	
Пункт 1.2. «Место учебной дисциплины в структуре программы подготовки специалистов среднего звена» представлен	+	
Пункт 1.3. «Цели и задачи учебной дисциплины – требования к результатам освоения учебной дисциплины» представлен	+	
Пункт 1.4. «Количество часов на освоение рабочей программы учебной дисциплины» представлен	+	
Объем максимальной учебной нагрузки обучающегося в паспорте программы представлен	+	
Объем обязательной аудиторной нагрузки в паспорте программы представлен	+	
Объем времени, отведенного на самостоятельную работу представлен	+	
Экспертиза Раздела 2. Структура и содержание учебной дисциплины		
Раздел 2. «Структура и содержание учебной дисциплины» представлен	+	
Таблица 2.1. «Объем учебной дисциплины и виды учебной работы» представлены	+	
Таблица 2.2. «Тематический план и содержание учебной дисциплины» представлены	+	
Уровни усвоения дидактических единиц представлены	+	
Объем максимальной учебной нагрузки обучающегося в паспорте программы и таблице 2.1 совпадает	+	
Объем обязательной аудиторной нагрузки в паспорте программы, таблицах 2.1 и 2.2 совпадает	+	
Объем времени, отведенного на самостоятельную работу обучающихся, в паспорте программы, таблицах 2.1 и 2.2 совпадает	+	
Экспертиза Раздела 3. Условия реализации учебной дисциплины		
Раздел 3 «Условия реализации программы дисциплины» представлен	+	
Пункт 3.1. «Требования к минимальному материально-техническому обеспечению» представлен	+	
Пункт 3.2. «Информационное обеспечение обучения» представлен	+	
Экспертиза Раздела 4. Контроль и оценка результатов освоения учебной дисциплины		
Раздел 4. «Контроль и оценка результатов освоения учебной дисциплины» представлен	+	
Перечень знаний и умений представлен	+	
Перечень форм и методов контроля и оценки результатов обучения представлен	+	
Программа дисциплины может быть направлена на содержательную экспертизу	+	

Эксперт:  (Кудряк О.А.), преподаватель высшей квалификационной категории
Дата «24» мая 2024 г.