Государственное бюджетное образовательное учреждение среднего профессионального образования (среднее специальное учебное заведение) «Южно-Уральский многопрофильный колледж»

|  |  |
| --- | --- |
|  | **СОГЛАСОВАНО**  Председатель ПЦК (МК)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись председателя ПЦК (МК)  «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_ 20\_\_\_г |

**Комплект**

**контрольно-оценочных средств**

**по учебной дисциплине**

**ОП.05. Основы программирования**

основной профессиональной образовательной программы (ОПОП)

по специальности СПО

**230115 Программирование в компьютерных системах**

Базовой подготовки

Челябинск, 2015

Разработчики:

|  |  |  |
| --- | --- | --- |
| ГБОУ СПО (ССУЗ) ЮУМК  ГБОУ СПО (ССУЗ) ЮУМК | преподаватель  преподаватель | Воропановаа И.О  Сухорослова Л.В. |
|  |  |  |

Эксперты:

|  |  |  |
| --- | --- | --- |
| ГБОУ СПО (ССУЗ) ЮУМК  ГБОУ СПО (ССУЗ) ЮУМК | преподаватель  преподаватель | Кранцевич Д.В.  Чухарев В.М. |
|  |  |  |

**Содержание**

|  |  |
| --- | --- |
| 1. Паспорт комплекта контрольно-оценочных средств | 3 |
| 1.1. Область применения комплекта контрольно-оценочных средств | 3 |
| 1.2. Система контроля и оценки освоения программы учебной дисциплины | 10 |
| 1.2.1. Формы промежуточной аттестации по учебной дисциплине | 10 |
| 1.2.2. Организация текущего контроля успеваемости, промежуточной аттестации по итогам освоения учебной дисциплины | 10 |
| 2. Задания для контроля и оценки освоения программы учебной  дисциплины | 11 |
| 2.1. Задания для текущего контроля | 11 |
| 2.2. Задания для промежуточной аттестации | 65 |
| 3. Рекомендуемая литература и иные источники | 82 |
|  |  |

**1. Паспорт комплекта контрольно-оценочных средств**

**1.1. Область применения комплекта контрольно-оценочных средств**

Комплект контрольно-оценочных средств, предназначен для проверки результатов освоения учебной дисциплины «Основы программирования» (далее УД) основной профессиональной образовательной программы (далее ОПОП) по специальности СПО 230115 Программирование в компьютерных системах

**Комплект контрольно-оценочных средств позволяет оценивать:**

1. Формирование элементов профессиональных компетенций (ПК) и элементов общих компетенций (ОК):

Таблица 1.

|  |  |  |
| --- | --- | --- |
| **Профессиональные и общие компетенции** | **Показатели оценки результата** | **Средства проверки**  **(№№ заданий)** |
| 1 | 2 | 3 |
| ПК 1.1. Выполнять разработку спецификаций отдельных компонент. | - четкость детализации отдельных компонентов задачи;  - полнота реализация математической модели в виде программного кода;  - правильность оформление документации на программные средства;  - оптимальное использование инструментальных средства для автоматизации оформления документации;  - применение алгоритмического языка для создания спецификаций;  - использование свойств алгоритма. | ПЗ№ 3 |
| П.К.1.2. Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля. | - создание математической модели;  - реализация всех этапов решения задачи на компьютере;  - использование трех типов управляющих структур;  - определение типов данных и переменных для них;  -использование технологии программирования «сверху вниз». | Практическое задание № 2 |
| ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств | - компиляция программного кода;  - устранение ошибок периода компиляции;  - оптимизация программного кода;  - демонстрация правильности программного кода на различных этапах решения задачи. | Практическое задание № 4-13 |
| ПК 1.4. Выполнять тестирование программных модулей. | - разработка корректных тестовых заданий и сценариев;  - тестирование методом «белого ящика»;  - тестирование методом «черного ящика». | Практическое задание № 4-13 |
| ПК 1.5. Осуществлять оптимизацию программного кода модуля | - выявление избыточности кода  программного продукта;  - определение полноты и отсутствие избыточности исходных текстов ПО на уровне функциональных объектов (процедур, функций);  - соответствие требованиям, предъявляемым к составу и содержанию, документации. | Практическое задание № 4-13 |
| **ПК 3.1.** Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения. | - умение читать блок-схемы;  - осуществление модернизации отдельных фрагментов программного кода;  - использование инструментальных  средств и графических языков  спецификаций для создания  компонент проектной и технической  документации;  - оформление проектной и технической  документации в соответствии со  стандартами. | Практическое задание № 4-13 |
| **ОК 1.** Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес | Проявление интереса к специальности через:  - участие во внеурочных организационных формах (олимпиадах, конференциях, неделях специальности и т.п.);  - изучение дополнительной профессиональной литературы для успешного овладения специальностью;  - систематическую подготовку к учебным занятиям и своевременное выполнение заданий в рамках внеаудиторной работы. | Ведомости, отражающие  положительную динамику  или стабильную  успеваемость по дисциплине. |
| **ОК 2.** Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество. | Рациональность планирования организации собственной деятельности в соответствии с профессиональными целями.  Объективность анализа профессиональной деятельности.  Аргументированная оценка качества выполнения профессиональных задач. |  |
| **ОК 3.** Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность | Результативность решения профессиональных проблем.  Оперативное решение нестандартных задач.  Анализ профессиональных ситуаций с позиции возможностей и ожидаемых рисков. |  |
| **ОК 4.** Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития | Оперативность поиска необходимой информации с использованием различных средств.  Обоснованный выбор оптимального состава источников информации для решения профессиональных задач и самообразования. |  |
| **ОК 5.** Использовать информационно-коммуникационные технологии в профессиональной деятельности. | Эффективное использование прикладного программного обеспечения, информационных ресурсов и возможностей сети Интернет в профессиональной деятельности. |  |
| **ОК 6.** Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями. | Результативное общение с коллегами, руководством, социальными партнерами.  Успешно применять на практике коммуникативные качества личности при общении с сокурсниками, педагогами, сотрудниками, руководством, работодателем.  Соблюдать принципы профессиональной этики. |  |
| **ОК 7.** Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий. | Проявление ответственности за работу подчиненных и результат выполнения заданий. |  |
| **ОК 8. .** Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации. | Обоснованная постановка цели, умение делать выбор и применять методы и способы решения профессиональных задач.  Участие в общественной деятельности (научно-практических конференциях, конкурсах по профилю специальности и т.п.) для личностного развития. |  |
| **ОК 9.** Ориентироваться в условиях частой смены технологий в профессиональной деятельности. | Изучение и анализ инноваций в области разработки технологических процессов профессиональной деятельности.  Результативное использование инновационные технологии в профессиональной деятельности. |  |
| **ОК 10.** Исполнять воинскую обязанность, в том числе с применением полученных профессиональных знаний (для юношей). | Понимание возможностей использования профессиональных знаний и умений во время исполнения воинской обязанности. |  |

2. Освоение умений и усвоение знаний

Таблица 2.

|  |  |  |
| --- | --- | --- |
| **Освоенные умения, усвоенные знания** | **Показатели оценки результата** | **№№ заданий**  **для проверки** |
| 1 | 2 | 3 |
| **У 1.** работать в среде программирования; | - создание дружественного интерфейса;  - производит отладку и компиляцию программы;  - тестирование построенных алгоритмов различными способами;  - исправление синтаксических ошибок в коде программы;  - исправление логических ошибок в коде программы. | Задания для текущего контроля, № 1, 2 |
| **У 2.** реализовывать построенные алгоритмы в виде программ на конкретном языке программирования; | - использование стандартных процедур и функций;  - запись математических выражений на конкретном языке программирования;  -целесообразность интерпретации построенного алгоритма на конкретном языке программирования. | Задания для текущего контроля, № 4-12 |
| **З1.** этапы решения задачи на компьютере; | - точно и полно использует технологическую цепочку решения задачи на компьютере;  - четко раскрывает каждый этап решения задач на компьютере;  - оценивает результативность и эффективность каждого этапа при решении задачи. | ТЗ 2.21 |
| **З2.** базовые конструкции изучаемых языков программирования; | -перечисляет и различает базовые конструкциями изучаемого языка программирования;  -использует различные типы данных;  - применяет необходимые стандартные процедуры и функции; | Задания для текущего контроля, №4-12 |
| **З**3**.** принципы структурного и модульного программирования; | - называет принципы структурного программирования;  - делает вывод об использовании той или иной структуры;  -идентифицирует входные и выходные данные;  - выбирает необходимые варианты нужной структуры;  -проектирует программный продукт в соответствии с принципами структурного программирования. | Практическая работа № 4-13 |
| **З**4**.** принципы объектно-ориентированного программирования | - называет принципы объектно-ориентированного программирования;  - использует их при создании программ;  - конструирует объекты используя понятия ООП. | Практическая работа № 3 |

**1.2 Система контроля и оценки освоения программы учебной дисциплины**

1.2.1. Формы промежуточной аттестации по УД

Таблица 3.

|  |  |
| --- | --- |
| **Учебная дисциплина** | **Формы промежуточной аттестации** |
| ОП.05 Основы программирования | экзамен |

1.2.2. Организация текущего контроля успеваемости, промежуточной аттестации по итогам освоения программы учебной дисциплины

Текущий контроль знаний и умений осуществляется по результатам устных ответов обучающегося, тестирования, выполнения практических заданий, в том числе обучающиеся выполняют задания внеаудиторных самостоятельных заданий (презентации, сообщения доклады).

Формой промежуточной аттестации по итогам усвоения программы дисциплины « Основы программирования» является экзамен.

Условием допуска обучающихся к экзамену является 60% выполнение всех практических заданий и тестирования, а также сдача отчетов по самостоятельной работе.

Экзамен носит комплексный характер и включает в себя два вида заданий: тестовое и практическое. На экзамен отводится 80 мин, из них 35 мин на тестовое задание и 45 мин на практическое задание.

Критерии оценивания:

Предлагаемые критерии носят рекомендательный характер:

- оценка «отлично» выставляется обучающемуся за работу, выполненную безошибочно, в полном объеме с учетом рациональности выбранных решений;

- оценка «хорошо» выставляется обучающемуся за работу, выполненную в полном объеме с недочетами;

- оценка «удовлетворительно» выставляется обучающемуся за работу, выполненную не в полном объеме (не менее 50% правильно выполненных заданий от общего объема работы);

- оценка «неудовлетворительно» выставляется обучающемуся за работу, выполненную не в полном объеме (менее 50% правильно выполненных заданий от общего объема работы);

2. Задания для контроля и оценки освоения программы учебной дисциплины

* 1. Задания для текущего контроля
     1. Примерные практические задания

**Практическое задание №1(Время выполнения 45 мин.)**

Запишите выражения на языке программирования.

Выражения приведены в таблице:

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 1+x+ | 9 |  |
| 2 |  | 10 |  |
| 3 |  | 11 |  |
| 4 |  | 12 |  |
| 5 |  | 13 |  |
| 6 |  | 14 |  |
| 7 |  | 15 |  |
| 8 |  | 16 |  |

**Практическое задание №2 «Стандартные процедуры и функции»**

**(время выполнения 30 мин. З2)**

1. Вычислите, используя операции div и mod:
2. 28 div 8
3. 28 mod 8
4. 76 div 22
5. 76 mod 22
6. 2222 div 10000
7. 2222 mod 10000
8. Определите значение функции для заданного целочисленного значения аргумента:
9. abs(l-52)
10. abs(32-33)
11. sqrt(25)
12. sqr(100-77)
13. Определите значение функции для заданного целочисленного значения аргумента
14. х=15, inс(х)
15. х=15, inc(х,2)
16. х=10000, dec(x)
17. х=33, dec(x, -33)
18. Определите значение функции для заданного вещественного значения аргумента:
19. trunc(100.9)
20. int(14.3)
21. round(0.09)
22. sqrt(0.0016)
23. Найдите значение выражения

2009 div 10 div 10 mod 100 - sqrt (81) \*int (2.3)

Критерии оценивания:

Предлагаемые критерии носят рекомендательный характер:

- оценка «отлично» выставляется обучающемуся за работу, выполненную безошибочно, в полном объеме с учетом рациональности выбранных решений;

- оценка «хорошо» выставляется обучающемуся за работу, выполненную в полном объеме с недочетами;

- оценка «удовлетворительно» выставляется обучающемуся за работу, выполненную не в полном объеме (не менее 50% правильно выполненных заданий от общего объема работы);

- оценка «неудовлетворительно» выставляется обучающемуся за работу, выполненную не в полном объеме (менее 50% правильно выполненных заданий от общего объема работы).

К практическим заданиям № 3-13 применяются критерии оценки программного продукта согласно показателям качества по ГОСТ 28195-89

|  |  |
| --- | --- |
| №  п\п | Показатели качества |
| 1 | Показатели надежности программного продукта:  - устойчивость функционирования  - работоспособность |
| 2 | Показатели сопровождения:  - структурность  - простота конструкции  - наглядность  - повторяемость |
| 3 | Показатели удобства применения  - легкость освоения  - доступность эксплуатационных программных документов  - удобство эксплуатации и обслуживания |

Критерии оценки:

|  |  |  |  |
| --- | --- | --- | --- |
| «5» | «4» | «3» | «2» |
| Созданный  программный  продукт  разработан в  полном  соответствии с  показателями  качества | Созданный  программный  продукт имеет  несоответствие  одному из  показателей. | Созданный  программный  продукт имеет  несоответствие по  двум показателям. | Созданный  программный  продукт не  соответствует  более 2  показателям. |

Примерный образец выполнения практического задания:

**Заданы координаты трех вершин треугольника А**(***x1,y1), B(x2,y2), C*(x3,y3). Найти его периметр и площадь.**

* **Построение математической модели:**

Пусть

L1- расстояние между точками А и В,

L2- расстояние между точками А и С,

L3- расстояние между точками С и В,

P-периметр треугольника, p-полупериметр,S-площадь, тогда

;

;

P=L1+L2+L3;

* **Блок-схема:**

начало

x1, y1, x2, y2, x3, y3

;

;

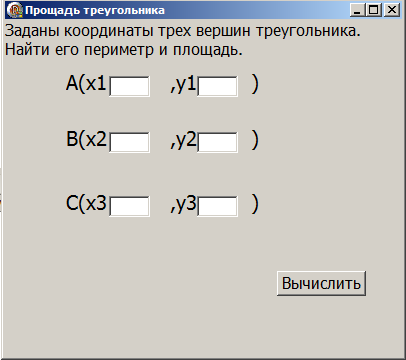
;

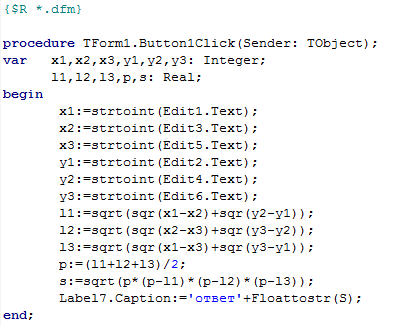
P=L1+L2+L3;

S, P

конец

* **Интерфейс приложения.**

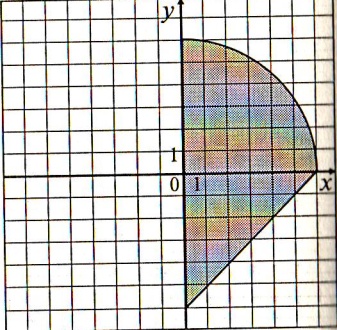
****

* **Программный код**

**Практическое задание № 3 «Линейные программы»**

1. Вычислить значение выражения по формуле (все переменные принимают действительные значения):
2. Заданы координаты трех вершин треугольника (*x1,y1), (x2,y2),* (x3,y3)- Найти его периметр и площадь.

**Практическое задание № 4 «Разветвляющиеся программы»**

1. Составить программу, печатающую значение true, если указанное высказывание является истиной, и false – в противном случае: Сумма цифр данного трехзначного числа N является четным числом.
2. Для данных областей составить программу, которая печатает принадлежит ли, если точка с координатами (х, у) закрашенной области, или нет
3. Составить программу, которая по данному числу (1—12) вы водит название соответствующего ему месяца.

**Практическое задание № 5 «Циклы»**

1. Одноклеточная амеба каждые 3 часа делится на 2 клетки. Определить, сколько амеб будет через 3, 6, 9, 12, 24 часа
2. Дано натуральное число N. Вычислить:
3. Даны числовой ряд и некоторое число ɛ. Найти сумму тех членов ряда, модуль которых больше или равен заданному ɛ. Общий член ряда имеет вид:
4. Составить программу для вычисления значений функции F(x) на отрезке [a, b] с шагом h. Результат представить в виде таблицы, первый столбец которой — значения аргумента, второй — соответствующие значения функции.

**Практическое задание № 6** «Подпрограммы»

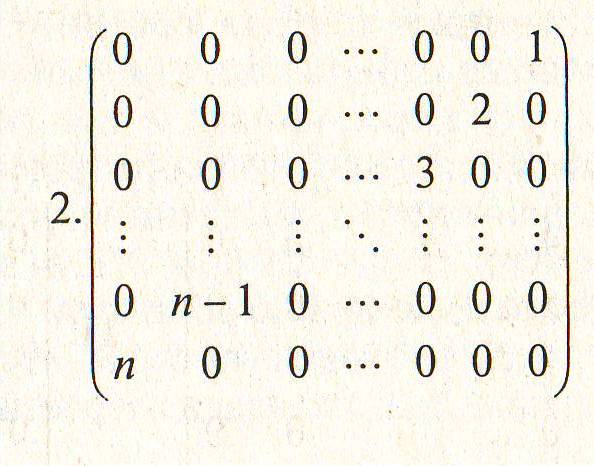
Написать программу, которая находит и выводит на печать все четырехзначные числа вида abcd, для которых выполняется: ab-cd = а + b + с + d.

**Практическое задание №7 «Строки»**

1. Дана строка, содержащая английский текст. Найти количество слов, начинающихся с буквы b.
2. Дана строка. Указать те слова, которые содержат хотя бы одну букву к.

**Практическое задание № 8 «Массивы»**

1. В целочисленной последовательности есть нулевые элементы. Создать массив из номеров этих элементов.
2. Дана последовательность действительных чисел *a1,* а2,..an. Указать те ее элементы, которые принадлежат отрезку [с, d].
3. Сформировать квадратную матрицу порядка n по заданному образцу:

**

**Практическое задание № 9 «Записи»**

Из данного списка спортсменов распечатать сведения о тех из них, кто занимается плаванием. Указать возраст, сколько лет они занимаются спортом.

**Практическое задание №10 «Множества»**

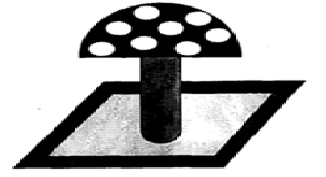
Подсчитать количество различных цифр в десятичной записи натурального числа.

**Практическое задание № 12 «Файлы»**

1. Заполнить файл последовательного доступа f целыми числами, полученными с помощью генератора случайных чисел, получить в файле g те компоненты файла f, которые являются четными.
2. Дан файл, содержащий текст, записанный строчными русскими буквами. Получить в другом файле тот же текст, записанный заглавными буквами.

**Практическое задание № 13 «Графика»**

Нарисуйте данный рисунок в цвете.



**Практическое задание № 14 «Объектно-ориентированное программирование»**

Определить объект TFish — аквариумная рыбка. Рыбка имеет координаты, скорость, размер, цвет, направление движения. Методами объекта являются:

• Init — устанавливает значения полей объекта и рисует рыбу на экране методом Draw.

• Draw — рисует рыбу в виде уголка с острием в точке Coord и направленного острием по ходу движения рыбы.

• Look — проверяет несколько точек на линии движения рыбы. Если хоть одна из них отличается по цвету от воды, возвращаются ее цвет и расстояние до рыбы.

• Run — перемещает рыбу в текущем направлении на расстояние, зависящее от текущей скорости рыбы. Иногда случайным образом меняет направление движения рыбы. Если рыба видит препятствие, направление движения меняется, пока препятствие не исчезнет из поля зрения рыбы.

* + 1. Примерные тестовые задания:

**Методика оценивания качества выполнения заданий**

**(В.П. Беспалько)**

Количественным критерием оценки правильности выполнения тестовых заданий служит коэффициент Ка, представляющий собой отношение количества правильно выполненных обучающимися существенных операций (А) к общему числу существенных операций теста (Р) Ка = А/Р

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ка | 1,0-0,9 | 0,89-0,8 | 0,79-0,7 | < 0,7 |
| отметка | 5 | 4 | 3 | 2 |

Выберите, на ваш взгляд, один правильный ответ и обведите их кружочком.

**Тест 1 (знания)**

**Вариант 1**

|  |  |  |
| --- | --- | --- |
| 1. | **Константы ─ это:**   1. величины, которые могут менять свое значение в процессе выполнения программы 2. величины, которые не могут менять своего значения в процессе выполнения программы 3. обозначают строки программы, на которые передается управление во время выполнение программы 4. Верного ответа нет   *Эталон ответа: б* | Р=4 |
| 2. | Язык программирования образуют три составляющие:   1. алфавит, орфография, диалектика; 2. алфавит, синтаксис, семантика; 3. переменные, процедуры, функции; 4. модули, описание, реализация   *Эталон ответа: б* | Р=4 |
| 3. | **Остаток от целочисленного деления можно выразить следующей функцией:**   1. A mod B 2. A div B 3. Abs (A)\*B 4. Верного ответа нет   *Эталон ответа: а* | Р=4 |
| 4. | Функция, позволяющая преобразовать значение целочисленного выражения в строку:   1. IntToStr(s:Integer):String; 2. StrToInt(s:String):Integer; 3. SrtToFloat(S:String):Real; 4. FloatToStr(a:Real): String.   *Эталон ответа: а* | Р=4 |
| 5 | **Тип переменных REAL это:**   * 1. целочисленный тип   2. логический тип   3. натуральный тип   4. верного ответа нет   *Эталон ответа: г* | Р=4 |
| 6 | **Раздел описания меток начинается со служебного слова:**   1. VAR 2. LABEL 3. TYPE 4. CONST   *Эталон ответа: б* | Р=4 |
| 7 | **Числа в языке Pascal различаются:**   1. как целые и вещественные; 2. как натуральные и вещественные; 3. как целые и иррациональные; 4. как целые и рациональные   *Эталон ответа: а* | Р=4 |
| 8 | **Операторы в программе на языке Pascal отделяются друг от друга:**   1. апострофом; 2. двоеточием; 3. пробелом; 4. точкой с запятой   *Эталон ответа: б* | Р=4 |
| 9 | Процедура, которая выводит окно с сообщением и командной кнопкой OK.   1. InputBox(Заголовок, Подсказка, Значение); 2. ShowMessage(S); 3. MessageDlg(s,t,b,h); 4. InputBox(f);   *Эталон ответа: б* | Р=4 |
| 10 | **Укажите последовательность команд, в результате выполнения которых значения переменных Х и У поменяются местами:**   1. B:=X; X:=Y; Y:=В; 2. X:=Y; Y:=X; 3. Y:=X; B:=X; X:=Y; 4. C:=X; X:=Y; X:=C.   *Эталон ответа: а* | Р=4 |
| 11 | **Раздел описания констант начинается со служебного слова:**   1. LABEL 2. VAR 3. TYPE 4. CONST   *Эталон ответа: в* | Р=4 |
| 12 | **Символьный тип данных описывается с помощью служебного слова**   1. Boolean 2. Char 3. Word 4. Real   *Эталон ответа: б* | Р=4 |
| 13 | **Раздел операторов начинается служебным словом:**   1. VAR 2. INTEGER 3. BEGIN 4. Верного ответа нет   *Эталон ответа: в* | Р=4 |
| 14 | **Алфавит языка программирования - это:**   1. Фиксированный набор символов, однозначно трактуемых; 2. А я; 3. a..z;   набор слов, которые понимает компьютер  *Эталон ответа: а* | Р=4 |
| 15 | **Целочисленное деление можно выразить следующей функцией:**   * 1. A mod B   2. A div B   3. Abs (A)\*B  1. Верного ответа нет   *Эталон ответа: б* | Р=4 |
| 16 | **Переменные – это:**   1. величины, которые могут менять свое значение в процессе выполнения программы 2. величины, которые не могут менять своего значения в процессе выполнения программы 3. обозначают строки программы, на которые передается управление во время выполнение программы 4. Верного ответа нет   *Эталон ответа: а* | Р=4 |

**Вариант 2**

|  |  |  |
| --- | --- | --- |
| 1. | **Язык программирования Pascal создал**:   1. Н. Вирт 2. Б. Паскаль 3. М. Фортран 4. Правильных ответов нет   *Эталон ответа: а* | Р=4 |
| 2. | **Раздел описания типов начинается со служебного слова:**   * 1. VAR   2. LABEL   3. TYPE   4. CONST   *Эталон ответа: в* | Р=4 |
| 3. | **В качестве имени в языке Pascal нельзя использовать:**   1. BR; 2. OR; 3. WR; 4. AR;   *Эталон ответа: б* | Р=4 |
| 4. | В результате выполнения функции на экране появляется диалоговое окно, в поле которого пользователь может ввести строку символов.   1. ShowMessage(S); 2. MessageDlg(s,t,b,h); 3. InputBox(Заголовок, Подсказка, Значение); 4. InputBox(f);   *Эталон ответа: в* | Р=4 |
| 5 | **Раздел VAR служит**   1. Для описания используемых переменных 2. Для описания величин 3. Для описания выражений 4. Верного ответа нет   *Эталон ответа: а* | Р=4 |
| 6 | **Логическая операция И**   1. or 2. end 3. else 4. and   *Эталон ответа: г* | Р=4 |
| 7 | **Выражение** - это :   1. конструкция языка, которая может меняться; 2. текс программы, заключенный в операторные скобки; 3. множество символов, которые является упорядоченными 4. конструкция, задающая правила вычисления значений переменных   *Эталон ответа: г* | Р=4 |
| 8 | **Тип переменных INTEGER это:**   * 1. целочисленный тип   2. логический тип   3. натуральный тип   4. верного ответа нет   *Эталон ответа: а* | Р=4 |
| 9 | **Раздел описания переменных начинается со служебного слова:**   1. VAR 2. LABEL 3. TYPE 4. CONST   *Эталон ответа: а* | Р=4 |
| 10 | **Комментарий к тексту программы на языке Pascal заключается:**   1. в фигурные скобки; 2. в круглые скобки; 3. в квадратные скобки; 4. между служебными словами Begin, End.   *Эталон ответа: а* | Р=4 |
| 11 | **Служебное слово Const в программе на языке Pascal фиксирует начало раздела программы, содержащего:**   1. описание переменных; 2. описание меток; 3. описание констант; 4. описание сложных типов данных.   *Эталон ответа: в* | Р=4 |
| 12 | Интервальный тип данных описывается:   1. Имя\_типа:=1,2,….,3,4… 2. Имя-типа=max..min; 3. Имя-типа:=min…max; 4. Имя-типа:=Min..max;   *Эталон ответа: г* | Р=4 |
| 13 | Язык программирования образуют три составляющие:   1. алфавит, орфография, диалектика; 2. алфавит, синтаксис, семантика; 3. переменные, процедуры, функции; 4. модули, описание, реализация.   *Эталон ответа: б* | Р=4 |
| 14 | **Логическая операция Или**   1. or 2. end 3. else 4. and   *Эталон ответа: а* | Р=4 |
| 15 | **Оператор присваивания имеет вид:**   * 1. =   2. :=   3. =:   4. Верного ответа нет   *Эталон ответа: б* | Р=4 |
| 16 | **Целочисленное деление можно выразить следующей функцией:**   1. A mod B 2. A div B 3. Abs (A)\*B 4. Верного ответа нет   *Эталон ответа: б* | Р=4 |

**Тест 2 (знания )**

**ВАРИАНТ1**

|  |  |  |
| --- | --- | --- |
| 1. | **Определите значение целочисленных переменных *x* и *y* после выполнения фрагмента программы:**  …  x: =8;  y: =4;  z: =y\*3;  x: =z div y mod x;  x:=x-1;  y:=y+1;  …   1. x=7,y=5 2. x=12,y=5 3. x=2,y=4 4. x=2,y=5   *Эталон ответа: г* | Р=4 |
| 2. | **Определите значение переменной *c* после выполнения фрагмента программы:**  …  a:=-6;  a:=a-6;  b:=-a;  c:=b-2\*b-2\*a;  …   1. c=-12 2. c=12 3. c=24 4. c=-24   *Эталон ответа: б* | Р=4 |
| 3. | **Определите значение целочисленных переменных *x,y* и *t* после выполнения фрагмента программы:**  …  x:=15;  y:=7;  t:=x div 3 mod y;  x:=y mod t;  y:=y-t;  …   1. x=2, y=2, t=2 2. x=5, y=2, t=2 3. x=2, y=5, t=2 4. x=2, y=2, t=5   *Эталон ответа: г* | Р=4 |
| 4. | **Определите значение вещественной переменной *a* после выполнения фрагмента программы:**  …  b:=7.53;  c:=7/2+2;  a:=trunc(b)+round(c);  …   1. a=12 2. a=9 3. a=13 4. a=10   *Эталон ответа: в* | Р=4 |
| 5 | **Определите значение вещественной переменной *a* после выполнения фрагмента программы:**  …  x:=123456789;  y:=x div 1000000 mod 10;  t:=x div 10000 mod 10;  x:=sqr(t mod y)+t mod 3;  a:=round(y/t)+exp((t-y)\*ln(x));  …   1. a=36 2. a=24 3. a=42 4. a=37   *Эталон ответа: г* | Р=4 |
| 6 | **Определите значение вещественных переменных *a*, *b* и целочисленной переменной *t*  после выполнения фрагмента программы:**  …  x:=17;  y:=3;  t:=y div (x+x mod y);  a:=trunc(x/y);  t:=t+t;  b:=int(a)+sqrt(t);  …   1. a=5,b=5,t=10 2. a=5,b=5,t=0 3. a=4,b=4,t=0 4. a=5,b=7,t= 4   *Эталон ответа: б* | Р=4 |
| 7 | **Определите значение целочисленных переменных *x,y* и z после выполнения фрагмента программы:**  …  X:=-5;  Y:=12;  Z:=x-y;  X:=abs (z) mod y;  Y:=sqr(x);  …   1. x=5, y=25, z=-17 2. x=5, y=25, z=17 3. x=7, y=49, z=7 4. x=7, y=14, z=-7   *Эталон ответа: а* | Р=4 |
| 8 | **Определите значение целочисленных переменных *a* и *b* после выполнения фрагмента программы:**  …  a:=sqr(15)\*6;  b:=(a div 10)+(a mod 100);  a:=(b div 10)\*(b mod 10);  …   1. a=90,b=185 2. a=3,b=13 3. a=185,b=90 4. a=13,b=3   *Эталон ответа: а* | Р=4 |
| 9 | **Определите значение переменной *c* после выполнения фрагмента программы:**  …  a:=sqr(7);  b:=a-(a mod 10);  c:=b-(a div b);  …   1. c=39 2. c=49 3. c=40 4. c=9   *Эталон ответа: а* | Р=4 |
| 10 | **Определите значение вещественных переменных *a* и *b* после выполнения фрагмента программы:**  …  a:=-6;  b:=8+3\*a;  b:=b/2\*a;  …   1. a=6,b=30 2. a=66,b=30 3. a=-6,b=30 4. a=66,b=1.2   *Эталон ответа: в* | Р=4 |
| 11 | **Определите значение вещественных переменных *a* и *b* после выполнения фрагмента программы:**  …  c:=2/3;  a:=round(c)+3\*c;  b:=trunc(c)+int(c);  …   1. a=3,b=2 2. a=0,b=3 3. a=2,b=3 4. a=3,b=0   *Эталон ответа: г* | Р=4 |
| 12 | **Определите значение вещественных переменных *a* и *b* после выполнения фрагмента программы:**  …  x:=987654321;  y:=x div 1000000 mod 10;  t:=x div 100 mod 10;  x:=sqrt(x)+int(y/t);  b:=exp(int(7/3)\*ln(t));  …   1. a=83,b=9 2. a=18,b=4 3. a=5,b=9 4. a=5,b=1   *Эталон ответа: в* | Р=4 |

Вариант 2

|  |  |  |
| --- | --- | --- |
| 1. | **Определите значение целочисленных переменных *x,y* и z после выполнения фрагмента программы:**  …  x:=13;  y:=3;  z:=x;  x:=z div y;  y:=x;  …   1. x=13, y=4, z=4 2. x=4, y=3, z=13 3. x=4, y=4, z=13 4. x=13, y=13, z=13   *Эталон ответа: в* | Р=4 |
| 2. | **Определите значение целочисленных переменных *a* и *b* после выполнения фрагмента программы:**  …  a:=sqr(15)\*6;  b:=(a div 10)+(a mod 10);  a:=(b div 10)\*(b mod 10);  …   1. a=90,b=185 2. a=3,b=13 3. a=185,b=90 4. a=13,b=3   *Эталон ответа: б* | Р=4 |
| 3. | **Определите значение целочисленных переменных *a* и *b* после выполнения фрагмента программы:**  …  a:=2010;  b:=2009;  a:=(a div 10) div 3;  b:=(b mod 1000)\*7;  …   1. a=66,b=46 2. a=63,b=67 3. a=67,b=63 4. a=66,b=63   *Эталон ответа: в* | Р=4 |
| 4. | **Определите значение вещественной переменной *a* после выполнения фрагмента программы:**  …  b:=7;  c:=b mod 3;  b:=-5;  a:=abs(b)+int(3.6)\*c;  …   1. a=8 2. a=11.0 3. a=-2 4. a=2   *Эталон ответа: а* | Р=4 |
| 5 | **Определите значение вещественных переменных *a* и *b* после выполнения фрагмента программы:**  …  a:=sqrt(9)+sqrt(9/25);  b:=int(a);  c:=a-b;  a:=round(c);  b:=b-a;  …   1. a=2,b=3 2. a=3,b=4 3. a=0,b=3 4. a=1,b=2   *Эталон ответа: г* | Р=4 |
| 6 | **Определите значение вещественных переменных *a* и *b* после выполнения фрагмента программы:**  …  c:=17/3;  a:=int(c)+trunc(c);  b:=sqrt(sqr(a));  …   1. a=10,b=1 2. a=10,b=100 3. a=9,b=9 4. a=9,b=81   *Эталон ответа: а* | Р=4 |
| 7 | **Определите значение целочисленных переменных *x* и *y* после выполнения фрагмента программы:**  …  x:=11;  y:=5;  t:=y;  y:=x mod y;  x:=t;  y:=y+2\*t;  …   1. x=5, y=10 2. x=10, y=15 3. x=5, y=11 4. x=11, y=5   *Эталон ответа: в* | Р=4 |
| 8 | **Определите значение целочисленных переменных *a* и *b* после выполнения фрагмента программы:**  …  a:=18\*5+28;  b:=(a div 10)-10;  a:=(b mod 10)+3;  …   1. a=4,b=10 2. a=0,b=1 3. a=1,b=11 4. a=4,b=1   *Эталон ответа: г* | Р=4 |
| 9 | **Определите значение целочисленных переменных *a* и *b* после выполнения фрагмента программы:**  …  c:=267348;  x:=c div 100000;  y:=c div 10000 mod 10;  d:=c div 10 mod 10;  a:=x+y;  x:=c mod 10;  y:=c div 1000 mod 10;  a:=a+y;  b:=d+x+(c div 100 mod 10);  …   1. a=15,b=15 2. a=10,b=10 3. a=13,b=13 4. a=16,b=18   *Эталон ответа: а* | Р=4 |
| 10 | **Определите значение вещественной переменной *a* после выполнения фрагмента программы:**  …  a:=8;  b:=a/2;  c:=sqrt(b);  a:=exp(c\*ln(b))+exp(1/3\*ln(a));  …   1. a=6 2. a=18 3. a=34 4. a=22   *Эталон ответа: б* | Р=4 |
| 11 | **Определите значение вещественных переменных *a*, *b* и целочисленной переменной *t*  после выполнения фрагмента программы:**  …  x:=9;  y:=3;  t:=y mod x;  b:=y;  a:=x;  a:=exp(3\*ln(int(t/a)+b)));  b:=power(, t);exp(t\*ln(sqrt(4)))  …   1. a=27,b=8,t=3 2. a=16,b=32,t=9 3. a=0,b=3,t=1 4. a=2,b=2,t=3   *Эталон ответа: а* | Р=4 |
| 12 | **Определите значение вещественных переменных *a*, *b* и целочисленной переменной *t*  после выполнения фрагмента программы:**  …  x:=14;  y:=3;  t:=y div x+x mod y;  a:=trunc(x/y);  t:=t+t;  b:=int(a)+sqrt(t);  …   1. a=4,b=4,t=4 2. a=4,b=6,t=4 3. a=4,b=4,t=6 4. a=6,b=4,t=4   *Эталон ответа: б* | Р=4 |

**Тест 3 (знания )**

Вариант 1

|  |  |  |
| --- | --- | --- |
| 1. | **Условный оператор применяется для программирования:**   * 1. Сложных алгоритмов   2. Линейных алгоритмов   3. Ветвящихся алгоритмов   4. Циклических алгоритмов   *Эталон ответа: в* | Р=4 |
| 2. | **Общий вид записи условного оператора в сокращенной форме**   * 1. case K of:   a1: <оператор 1>;  a2: <оператор 2>;  …  an: <оператор n>  Else <оператор n+1>  End;   * 1. case K of:   a1: <оператор 1>;  a2: <оператор 2>;  …  an: <оператор n>  Else <оператор n+1>  End   * 1. If <оператор 1> then <оператор 2 > else <оператор 3>:   2. If <оператор 1> then <оператор 2 >   *Эталон ответа: г* | Р=4 |
| 3. | **Результатом выполнения фрагмента алгоритма будет следующее значение k**  …  a:=7;  b:=8;  k:=5+a;  if a>b then k:=k\*3  else k:=k-3;  …   * 1. *7*   2. 8   3. 9   4. Верного ответа нет   *Эталон ответа: в* | Р=4 |
| 4. | **Оператор CASE служит для создания:**   * 1. Линейных алгоритмов   2. Ветвящихся алгоритмов   3. Циклических алгоритмов   4. Верного ответа нет   *Эталон ответа: б* | Р=4 |
| 5 | **В каком из условных операторов допущена синтаксическая ошибка:**   * 1. If B=0 then label1.Caption:=‘ Деление на нуль невозможно’;   2. If a>b then max := a else max:=b;   3. If (a>b) and (b>0) then c:=a+b;   4. If a<b then min:=a; else min:=b;   *Эталон ответа: г* | Р=4 |
| 6 | **Можно ли в условном операторе использовать более одного оператора?**   1. да 2. нет 3. только один 4. нет верного ответа   *Эталон ответа: б* | Р=4 |
| 7 | **Любое логическое выражение, которое записывается с помощью знаков сравнения (<, >, =) называется …**   1. условие 2. Программа 3. составной оператор 4. инверсия   *Эталон ответа: а* | Р=4 |
| 8 | **Как обозначается логическое отрицание (инверсия)?**   1. then 2. and 3. or 4. not   *Эталон ответа: г* | Р=4 |
| 9 | **Служебное слово IF в условном операторе переводится как...**   1. если 2. тогда 3. иначе 4. условие   *Эталон ответа: а* | Р=4 |
| 10 | **Укажите правильный вариант использования условного оператора**.   1. IF <условие> THEN begin <серия операторов 1> end ELSE begin <серия операторов 2> end; 2. IF <условие> THEN <серия операторов 1>; ELSE <серия операторов 2>; 3. THEN begin <серия операторов 1> end IF <условие>  ELSE begin <серия операторов 2> end; 4. IF <условие> THEN <серия операторов 1>   *Эталон ответа: а* | Р=4 |
| 11 | **Какое по типу выражение вычисляется в условии оператора if**   1. числовое 2. логическое; 3. неравенство 4. равенство   *Эталон ответа: б* | Р=4 |
| 12 | **Что будет выведено на экран после выполнения фрагмента программы:**  …  x:=0;  if x<>5 then y:=10 else y:=0;  label1.caption:=inttostr(y);  …   1. 10 2. 5 3. ничего 4. y   *Эталон ответа: а* | Р=4 |
| 13 | **Что будет выведено на экран после выполнения фрагмента программы:**  …  a:=12;  if a>10 then n:=a+5 else n:=a-1;  label1.caption:=inttostr(n);  …   1. 17 2. 11 3. 13 4. 10   *Эталон ответа: а* | Р=4 |
| 14 | **Что будет выведено на экран после выполнения фрагмента программы:**  …  x:=0; y:=0;  if (x=0) or (y<>0) then n:=7 else n:=11;  label1.caption:=inttostr(n);  …   1. 11 2. 7 3. 0 4. n   *Эталон ответа: б* | Р=4 |
| 15 | **Что будет выведено на экран после выполнения фрагмента программы:**  …  M:=5;  If m>5 then y:=5+m else y:=100-m;  label1. caption:=inttostr(y);  …   1. 95 2. 10 3. 5 4. 0   *Эталон ответа: а* | Р=4 |
| 16 | **Что будет выведено на экран после выполнения фрагмента программы:**  …  a:=1;  if (a=1) then n:=a else n:=-a;  label1.caption:=inttostr(n);  …   1. 0 2. 1 3. -1 4. -2   *Эталон ответа: б* | Р=4 |
| 17 | **Какое условие следует использовать для вычисления Y по формулам:**   1. if M=0 then N:=sin(M) else N:=M+cos(M); 2. if M<>0 then N:=sin(M) else N:=M+cos(M); 3. if M<>0 then N:=sin(M)+cos(M) else N:=M+cos(M); 4. if M<>0 then N:=sin(M) ;   *Эталон ответа: а* | Р=4 |
| 18 | **Цикл WHILE называется :**   * 1. Цикл с предусловием   2. Цикл с параметром   3. Цикл с постусловием   4. Ветвлением   *Эталон ответа: а* | Р=4 |
| 19 | **Тип переменных для параметра цикла FOR:**   * 1. целочисленный тип   2. логический тип   3. натуральный тип   4. верного ответа нет   *Эталон ответа: а* | Р=4 |
| 20 | **Оператором цикла не является оператор**   * 1. While…   2. Case…   3. Repeat…   4. Верного ответа нет   *Эталон ответа: б* | Р=4 |
| 21 | **Сколько раз будет выполняться цикл For i:=7 to 12 do…;**   1. 5 раз; 2. 7 раз; 3. 6 раз; 4. 4 раза.   *Эталон ответа: в* | Р=4 |
| 22 | **Какой из циклов не содержит логического условия:**   * 1. While   2. For   3. Repeat   4. Все содержат   *Эталон ответа: б* | Р=4 |
| 23 | **Требуется подсчитать сумму натуральных чисел от 5 до 125. Какое условие нужно использовать в цикле While?**   1. i>125; 2. i<125; 3. i<=124; 4. i<=125.   *Эталон ответа: б* | Р=4 |
| 24 | **Выберите цикл с постусловием:**   * 1. While   2. For   3. Repeat   4. Case   *Эталон ответа: в* | Р=4 |
| 25 | 1. **Какой цикл изображен на блок схеме?**   Рисунок1   * 1. While…;   2. Repeat…;   3. For … to…   4. For … DownTo…   *Эталон ответа: б* | Р=4 |

Вариант 2

|  |  |  |
| --- | --- | --- |
| 1. | **Ветвящийся алгоритм подразумевает:**   * 1. Неоднократное повторение отдельных частей программы   2. Последовательное выполнение всех элементов программы   3. Выполнение лишь нескольких, удовлетворяющих заданному условию частей программы   4. Переход в указанное место программы   *Эталон ответа: в* | Р=4 |
| 2. | **Чему будет равно значение переменной М в результате выполнения серии операторов**  N:=5;  Y:= 9 mod 4;  If Y>0 then M:=N/5+3\*Y\*N   * 1. 31   2. 16   3. 0   4. 5   *Эталон ответа: б* | Р=4 |
| 3. | **Общий вид записи условного оператора в полной форме**   1. case K of:   a1: <оператор 1>;  a2: <оператор 2>;  …  an: <оператор n>  Else <оператор n+1>  End;   1. case K of:   a1: <оператор 1>;  a2: <оператор 2>;  …  an: <оператор n>  Else <оператор n+1>  End   1. If <оператор 1> then <оператор 2 > else <оператор 3>: 2. If <оператор 1> then <оператор 2 >   *Эталон ответа: в* | Р=4 |
| 4. | **Как запишется условный оператор для поиска большего из двух чисел?**   1. if a>b else max:=a then max:=b; 2. if a>b then max:=a else max:=b; 3. a>b if max:=a then max:=b else; 4. if a>b then max=a else max=b;   *Эталон ответа: б* | Р=4 |
| 5 | **Как обозначается логическое умножение (конъюнкция)?**   1. then 2. And 3. Or 4. No   *Эталон ответа: в* | Р=4 |
| 6 | **Служебное слово THEN в условном операторе переводится как...**   1. Если 2. Тогда 3. Иначе 4. Ввод   *Эталон ответа: а* | Р=4 |
| 7 | **Служебное слово ELSE в условном операторе переводится как...**   1. Если 2. Тогда 3. Иначе 4. Условие   *Эталон ответа: в* | Р=4 |
| 8 | **Укажите правильный вариант использования неполного ветвления.**   1. IF <условие> THEN begin <серия операторов 1> end; 2. THEN <серия операторов 1>; ELSE <серия операторов 2>; 3. ELSE  <серия операторов 2>  IF <условие>; 4. THEN  <условие> IF <серия операторов 1>   *Эталон ответа: а* | Р=4 |
| 9 | **Что будет выведено на экран после выполнения фрагмента программы:**  …  a:=3;  if a>3 then a:=5 else a:=1;  writeln (a);  …   1. 1 2. 3 3. 5 4. ничего   *Эталон ответа: а* | Р=4 |
| 10 | **Что будет выведено на экран после выполнения фрагмента программы:**  …  a:=1; b:=2;  if (a>=1) and (b>=2) then n:=a+b else n:=a-b;  writeln (n);  …   1. 3 2. -1 3. 2 4. 0   *Эталон ответа: а* | Р=4 |
| 11 | **Какое условие следует использовать для вычисления Y по формулам:**  *Y = | X + 1 |, при X<1;*  *Y= cos X, при X>=1;*   1. if X>=1 then Y:=cos(X) else Y:=abs(X+1); 2. if X<1 then Y:=cos(X) else Y:=abs(X+1); 3. if X>=1 then Y:=abs(X+1) else Y:=cos(X); 4. if X=1 then Y:=cos(X) else Y:=abs(X+1);   *Эталон ответа: а* | Р=4 |
| 12 | **Какое условие следует использовать для вычисления Y по формулам:**   1. if X>0 then Y:=X+15 else Y:=X-15; 2. if X>0 then Y:=X-15 else Y:=X+15; 3. if X<0 then Y:=X+15 else Y:=X-15; 4. if X<=0 then Y:=X+15;   *Эталон ответа: а* | Р=4 |
| 13 | **Входит ли число R в диапазон значений [-5 ; 0).**  **Какое составное условие нужно поставить в этой задаче?**   1. (R<=-5) and (R>0) 2. (R>-5) and (R<=0) 3. (R>=-5) and (R<0) 4. (R-5) and (R+0)   *Эталон ответа: в* | Р=4 |
| 14 | **В каком из условных операторов допущена синтаксическая ошибка:**   * 1. If B=0 then label1.Caption:=‘ Деление на нуль невозможно’;   2. If a>b then max := a else max:=b;   3. If (a>b) and (b>0) then c:=a+b;   4. If a<b then min:=a; else min:=b**;**   *Эталон ответа: г* | Р=4 |
| 15 | Альтернатива - это:  а) композиция разных действий;  б) вариант;  в) конструкция ветвления;  г) шаг выполнения программы  *Эталон ответа: в* | Р=4 |
| 16 | Переменная логического типа может принимать значения   1. Ложь и истина 2. 1 и 0 3. False и true 4. False и true   *Эталон ответа: в* | Р=4 |
| 17 | **Цикл REPEAT называется**   * 1. Цикл с предусловием   2. Цикл с параметром   3. Цикл с постусловием   4. Ветвлением   *Эталон ответа: в* | Р=4 |
| 18 | **Сколько раз будет выполняться цикл For i:=4 to 8 do…;**   * 1. 5 раз;   2. 6 раз;   3. 8 раз;   4. 4 раза.   *Эталон ответа: а* | Р=4 |
| 19 | **Какой из операторов цикла имеет 2 вида записи**   1. For… 2. While… 3. Repeat… 4. Верного ответа нет   *Эталон ответа: а* | Р=4 |
| 20 | **Какой цикл изображен на блок схеме?**  Рисунок2   1. Repeat; 2. For ; 3. While 4. Верного ответа нет.   *Эталон ответа: в* | Р=4 |
| 21 | **Как исправить ошибку в операторе For для того чтобы вывести числа 10 9 8 7 …. -9 -10?**  **…**  **For i:=10 to -10**  **Do Label1.Caption:=IntoStr(i);**   1. ошибок нет, исправлять ни чего не надо; 2. 1-я строка должна выглядеть так: For i:=-10 to 10; 3. 1-я строка должна выглядеть так: For i:=10 Downto -10 4. 1-я строка должна выглядеть так: For i:=(-10 ) Downto 10.   *Эталон ответа: в* | Р=4 |
| 22 | **Цикл For называется:**   1. Цикл с предусловием 2. Цикл с параметром 3. Цикл с постусловием 4. Ветвлением   *Эталон ответа: б* | Р=4 |
| 23 | **Требуется подсчитать сумму натуральных чисел от 2 до 22. Какое условие нужно использовать в цикле While?**   * 1. i<23;   2. i>22;   3. i=22;   4. i>=22   *Эталон ответа: а* | Р=4 |
| 24 | **В каком из операторов допущена смысловая ошибка:**   * 1. For i:=1 to 20 do p:=p+1;   2. While s<3 do s:=s-3;   3. Repeat k<1 until k:=k+1;   4. For I:=10 downto 5 do p:=p+1;   *Эталон ответа: в* | Р=4 |
| 25 | **Чему будет равна переменная sum после выполнения фрагмента программы:**  **sum:=0; for i:=5 to 8 do**  **sum:=sum+i;**   * 1. 18;   2. 13;   3. 20;   4. 26.   *Эталон ответа: г* | Р=4 |

Вариант 1(знания )

|  |  |  |
| --- | --- | --- |
| 1. | **Чему будут равны значения строковых переменных st и stl после выполнения следующего фрагмента программы**?  **…**  St:='CDEFGHIJK';  Stl:='ABCD';  Stl:=Copy(St,2,3);  …   1. St='CDEFGHIJK', Stl=’DEF'; 2. St='CDEFGHIJK' Stl='EFG'; 3. St='CDEFGHIJK', Stl='EF'; 4. St='CDEFGHIJK', Stl='FG';   *Эталон ответа: б* | Р=4 |
| 2. | **Каков будет результат выполнения фрагмента программы?**  **Var S1:string;**  **…**  **S1:=Copy('крокодил',4,3);**  …   1. крок; 2. одил; 3. кродил; 4. код.   *Эталон ответа: г* | Р=4 |
| 3. | **Задана строка St. Определить, что выполняет фрагмент алгоритма**  …  **S:=0;**  **For i := 1 to Length (St) do**  **Begin**  **Var(St[i], d, k);**  **If k=0 then S:=S+d;**  **End;**  ***…***   1. определяет количество цифр в строке; 2. подсчитывает количество нулей  в строке; 3. определяет сумму номеров позиции в строке, где стоят цифры; 4. подсчитывает сумму цифр в строке.   *Эталон ответа: г* | Р=4 |
| 4. | **Чему будут равны значения строковых переменных** St **и St1 после выполнения следующего фрагмента программы?**  …  St:='PRQS';  Stl:='YZA';  Insert(Stl,St,3);  …   1. St='PRYZAQS' , Stl='YZA'; 2. St='PRQS', stl=' YZPRQSA'. 3. St='PRQS', Stl='YZA'; 4. St='YZA', stl='PRQS'.   *Эталон ответа: б* | Р=4 |
| 5 | **Чему будут равны значения переменных Y и K после следующего фрагмента программы?**  **Var Y:Real;**  **К:Integer;**  **St:String;**  **…**  St**:='9,3126';**  **Val(St, Y, K) ;**  …   1. Y=9.0, K=3126; 2. Y=0.3126, К=9; 3. y=9.0, К=2; 4. Y=9.3126, K=0;   *Эталон ответа: г* | Р=4 |
| 6 | **Каков будет результат выполнения фрагмента программы?**  **Var S1:string;**  **…**  **S1:='информатика';**  **Delete(s1,2,4);**  …   1. иматика; 2. форма; 3. инф; 4. инфо.   *Эталон ответа: а* | Р=4 |
| 7 | **После выполнения фрагмента программы определить значение переменной Х.**  **…**  **S:=’Учиться, учиться и еще раз учиться’;**  **X:=Pos(‘уч’,S);**  **…**   1. учится; 2. уч; 3. 1; 4. True   *Эталон ответа: в* | Р=4 |
| 8 | **Каков будет результат выполнения следующего фрагмента программы?**  **…**  **Var A:Array [1..10] of Integer;**  **Q, P:Integer;**  **i:integer ;**  **…**  **Q:=A[1];**  **P:=A[1];**  **For i:=1 To 10 Do**  **Begin**  **If A[i ]<P Then P:=A[i ];**  **If A[i]>Q Then Q:=A[i];**  **End;**  a) в переменных Q и Р будут содержаться непредсказуемые значения;  б) в переменной Q будет содержаться значение максимального элемента массива А, а в переменной P – значение минимального элемента;  в) в переменной Q будет содержаться значение минимального элемента массива А, а в переменной Р – значение максимального элемента;  г) фрагмент содержит ошибки и работать не будет.  *Эталон ответа: б* | Р=4 |
| 9 | **Какой фрагмент программы формирует массив В, содержащий только отрицательные элементы массива А?**  а) …  For i:=1 To 10 Do  If A[i]<0 Then B[i]:=A[i];  …б) …  j:=1;  For i:=1 To 10 Do  Begin  If A[i]<0 Then B[j]:=A[i];  J:=j+1;  End;  в) …  J:=0;  For i:=1 To 10 Do  If A[i]<0 Then  begin  J:=j+1;  B[J]:=A[i];  End;  …  г) правильного варианта нет  *Эталон ответа: в* | Р=4 |
| 10 | **Дан массив А=(-5 6 0 4 0 -7).Что будет сформировано в переменной х после выполнения следующего фрагмента программы?**  …  **Х:=1;**  **For i:=1 To 6 Do**  **If A[i]>0 Then X:=X\*A[i];**  …  a) произведение отрицательных элементов (х=35);  б) произведение положительных элементов (х=24);  в) произведение всех элементов (х=0)  г) фрагмент содержит ошибки и работать не будет.  *Эталон ответа: б* | Р=4 |
| 11 | **Что делает следующий фрагмент программы с массивом A, содержащим 10 элементов?**  …  **M:=A[1];**  **For i:=1 To 9 Do A[i]:=A[i+1];**  **A[10]:=M;**  **…**   * + - 1. присваивает каждому элементу массива значение следующего элемента, а последнему – значение 1-го элемента;       2. присваивает каждому элементу массива значение предыдущего элемента, а последнему – значение 1-го элемента;       3. каждый элемент массива увеличивает на 1;       4. фрагмент содержит ошибки и работать не будет.   *Эталон ответа: в* | Р=4 |
| 12 | **Элемент матрицы A[4,2] находится на пересечении:**   * 1. 2-й строки и 4-го столбца;   2. 4-ой строки и 2-го столбца;   3. правильного ответа нет.   *Эталон ответа: б* | Р=3 |
| 13 | **Что определяет для массива X[1..n, 1..m] следующий алгоритм**  **…**  **For i:= 1 to n do**  **For j:= 1 to m do**  **If i mod 2 = 0 then X[i, j] = A;**  **…**   1. четные столбцы матрицы заменить на А; 2. четные строки матрицы заменить на А; 3. количество четных элементов массива; 4. нечетные строки матрицы заменить на А.   *Эталон ответа: в* | Р=4 |
| 14 | **Что определяет для массивов X[1..n, 1..m] и Y[1..n, 1..m] следующий алгоритм**  **…**  **For i:=1 to n do**  **For j:=1 to m do**  **X[i, j]: = X[i, j] + Y[i, j];**  **…**   1. сумма элементов главной диагонали матрицы; 2. сумма элементов каждой строки матриц X и Y; 3. четные строки матрицы заменить на А; 4. сумма матриц X и Y.   *Эталон ответа: г* | Р=4 |
| 15 | **В каком диапазоне будут находиться значения элементов матрицы А:**  …  **For i:=1 To 3**  **Do For j:=1 to 5**  **Do A[I,j]:=random(300)-200;**  …   * 1. От 0 до 300;   2. От 0 до 200;   3. От -200 до 99   4. Правильного ответа нет.   *Эталон ответа: в* | Р=4 |
| 16 | **Выберите правильный вариант объявления записи из четырёх полей:**  а) Var Z:array[1..4] of Integer;  б) Var Z:Record  Tovar:String[20];  Price:Real;  Nal:Integer;  End;  в) Var Z:Array[1..4] of (string of Real or Integer);  г) правильного варианта нет  *Эталон ответа: б* | Р=4 |
| 17 | **Как обратится к полю записи**   1. Имя\_поля:=значение; 2. Имя\_поля.запись:=значение; 3. Переменная.имя\_поля:=значение; 4. правильного варианта нет   *Эталон ответа: в* | Р=4 |
| 18 | **Запись – это**   1. константное значение; 2. последовательность, состоящая из фиксированного числа однотипных элементов; 3. последовательность однотипных элементов, отличающихся индексами; 4. последовательность, состоящая из фиксированного числа величин, называемых полями;   *Эталон ответа: г* | Р=4 |
| 19 | **Как правильно ввести значение полей записи с клавиатуры?**  **Var Z:Record**  **Tovar:String[20];**  **Price:Real;**  **Nal:Integer;**  **End;**  а) With Z Do  Begin  Tovar :=inputBox(‘Заполнение записи’,’ Название товара?’,’‘);  Price:=strToFloat(inputBox(‘Заполнение записи’, ’Цена?’,’ ‘));  Nal:=StrToInt(inputBox(‘Заполнение записи’, ’Наличие на складе (0/1)’,’ ‘));  End;  б) Z:=inputBox(‘Заполнение записи’,’ Название товара?, Цена?, Наличие на складе (0/1)’, ‘’)  в) Tovar :=inputBox(‘Заполнение записи’,’ Название товара?’, ’‘);  Price:=strToFloat(inputBox(‘Заполнение записи’, ’Цена?’,’ ‘));  Nal:=StrToInt(inputBox(‘Заполнение записи’, ’Наличие на складе (0/1)’,’ ‘));  г) правильного варианта нет.  *Эталон ответа: б* | Р=4 |
| 20 | **Конструктором множества называют**   1. Перечисление элементов множества; 2. Перечисление элементов множества через запятую в квадратных скобках; 3. Имя\_множества:=конструктор\_множества; 4. Перечисление элементов множества через запятую в круглых скобках;   *Эталон ответа: б* | Р=4 |
| 21 | **Какая операция над множествами показана?**  А  B   1. Объединение множеств; 2. Пересечение множеств; 3. Разность (В-А) множеств 4. Разность (А-В) множеств   *Эталон ответа: г* | Р=4 |
| 22 | **Что делает следующий фрагмент программы?**  **Type chisla= set of byte; Var z,x,y: chisla; …**  **x:= [0..150]; y:= [100..255]; z:= x\*y**  **…**   1. получено множества чисел из диапазона 100..150 2. получено множества чисел из диапазона 0..255 3. получено множества чисел из диапазона 150..255 4. получено множества чисел из диапазона 0..100   *Эталон ответа: а* | Р=4 |
| 23 | **Заполните пробелы в следующем предложении : При завершении работы с файлом его необходимо закрыть командой**   1. Eoln(файлов\_пер); 2. FileSize(файлов\_пер); 3. CloseFile(файлов\_пер); 4. AssignFile(файлов\_пер, ′путь к файлу′);   *Эталон ответа: в* | Р=4 |
| 24 | **Укажите, как будет записана команда, выполняющая следующие действия: Процедура ставит в соответствие файловую переменную ft и файл dosname.txt**   1. Write(ft, ′dosname.txt′); 2. ReadLn(ft, ′dosname.txt′); 3. AssignFile(ft, ′dosname.txt′); 4. Write(′dosname.txt′, ft);   *Эталон ответа: в* | Р=4 |
| 25 | **Заполните пробелы в следующем предложении. Для использования файла в режиме дозаписи необходима…**   1. Append(файлов\_пер); 2. Reset(файлов\_пер); 3. AssignFile(файлов\_пер, ′путь к файлу′); 4. FileSize(файлов\_пер);   *Эталон ответа: а* | Р=4 |

Вариант 2(знания)

|  |  |  |
| --- | --- | --- |
| 1. | **Чему будет равно значение строковой переменной** St **после выполнения следующего фрагмента программы?**  **…**  **St:='YZA';**  Delete (St,2,2);  …   1. St='YZA'; 2. st='Y'; 3. st='YZ'; 4. St= 'A'.   *Эталон ответа: б* | Р=4 |
| 2. | **Чему будет равно значение целой переменной К после выполнении следующего фрагмента программы?**  Var St,St1:String;  …  St:='ABCDEFGHIJK';  Stl:='KJI';  K:=Pos(Stl,St);  …   1. K=9; 2. K=11; 3. К=10; 4. К=0.   *Эталон ответа: г* | Р=4 |
| 3. | **После выполнения фрагмента программы определить значение переменной S.**  …  **S:=**'**мама**'**;**  **S1:=**'**мыла**'**;**  **S2:=**'**раму**';  **S3:=Copy (S,3,2)+Copy (S1,3,2);**  **S:=**'**Ми**'+**copy(S1,3,2)+**'**б**'**+Copy (S1,2,3)+S3;**  **…**   1. раму мыла мама; 2. Мила была мама; 3. Мила была мала; 4. Мама.   *Эталон ответа: в* | Р=4 |
| 4. | **Чему будут равны значения переменной S после выполнения следующего фрагмента программы?**  Var S:String[20];  …  S:=’форма’;  S[6]:=’т’;  …   1. форма; 2. фор; 3. формат; 4. ттттт.   *Эталон ответа: в* | Р=4 |
| 5 | **Чему будет равно значение целой переменной S после выполнении следующего фрагмента программы?**  **Var S:byte;**  **…**  **S:=length(' каникулы');**  **…**   1. S=0 2. S=l 3. S=8 4. S=true   *Эталон ответа: в* | Р=4 |
| 6 | **После выполнения фрагмента программы на экран будет выведено**  **…**  **S:=**'**учим Pascal**';  **For i:=1 to length(S) do**  **S[i]:=Upcase(S[i]);**   1. *'*учим Pascal*'*; 2. *'*УЧИМ pascal*'*; 3. *'*учим PASCAL*'*; 4. *'*учим pascal*'.*   *Эталон ответа: в* | Р=4 |
| 7 | **Дан массив А=(-5 6 0 4 0 -7).Что будет сформировано в переменной х после выполнения следующего фрагмента программы?**  **…**  **Х:=0;**  **For i:=1 To 6 Do**  **If A[i]>0 Then X:=X+A[i];**  …   1. сумма отрицательных элементов целочисленного массива А (х=-12); 2. сумма положительных элементов целочисленного массива А (х=10); 3. количество нулевых элементов (Х=2); 4. сумма всех элементов массива А (х=-2).   *Эталон ответа: б* | Р=4 |
| 8 | **Дан массив А=(-9 0 2 -6 5 4).Что будет сформировано в переменной х после выполнения следующего фрагмента программы?**  …  **Х:=0;**  **For i:=1 To 6 Do**  **If (А[i]<0) And (A[i] mod 2 <> 0) Then X:=X+A[i];**  …   1. сумма отрицательных нечетных элементов целочисленного массива А (х=-9); 2. произведение положительных четных элементов целочисленного массива А (х=8); 3. сумма положительных четных элементов целочисленного массива А (х=-9); 4. количество ненулевых элементов массива А (х=5).   *Эталон ответа: а* | Р=4 |
| 9 | **Дан массив А=(-5 6 0 4 0 -7).Что будет сформировано в переменной х после выполнения следующего фрагмента программы?**  …  **Х:=0;**  **For i:=1 To 6 Do**  **If A[i]<>0 Then X:=X+1;**  …   1. количество ненулевых элементов (х=4); 2. количество отрицательных элементов (х=2); 3. сумма всех элементов массива А (х=-2); 4. количество положительных элементов (х=2).   *Эталон ответа: а* | Р=4 |
| 10 | **Что делает следующий фрагмент программы с массивом А, содержащим N элементов?**  …  **For i:=1 To N div 2**  **Do Begin**  **M:=A[i];**  **A[i]:=A[N+1-i];**  **A[N+1-i]:=M;**  **End;**  …   1. присваивает каждому элементу массива значение следующего элемента, а последнему – значение 1-го элемента; 2. записывает элементы массива в обратном порядке; 3. присваивает каждому элементу массива значение предыдущего элемента, а последнему – значение 1-го элемента; 4. фрагмент содержит ошибки и работать не будет.   *Эталон ответа: б* | Р=4 |
| 11 | **Укажите правильное описание двумерного массива:**   1. a: array [1…n, 1…n] of real; 2. a: array {1..n,1..m} of integer; 3. a: array [1..n, 1..m] of real; 4. a: aray  [1..n, 1..m]: integer.   *Эталон ответа: в* | Р=4 |
| 12 | **Что определяет для массива X[1..m, 1..m] следующий алгоритм**  **…**  **For i:=1 to m do**  **For j:=1 to n do**  **If i = j then X[i, j] =1 else X[I,j]:=0;**  **…**   1. формирование матрицы, в которой элементы, стоящие на главной диагонали равны 1, остальные равны 0; 2. формирование матрицы, в которой элементы, стоящие в четных строках равны 1, остальные равны 0; 3. количество равных элементов матрицы X; 4. формирование матрицы, в которой элементы, стоящие на побочной диагонали равны 1, остальные равны 0.   *Эталон ответа: а* | Р=4 |
| 13 | **Что делает следующий фрагмент программы, если дана матрица H размером 12 x 5?**  …  **For j:=1 To 5 Do**  **Begin**  **P:=1;**  **For i:=2 to 12 Do**  **P:=P\*H[I,j];**  **K[j]:=P;**  **End;**  **…**   1. Присваивает переменной P значение произведений всех элементов матрицы H; 2. Формирует массив k из произведений всех элементов каждого столбца матрицы H; 3. Фрагмент содержит ошибку и работать не будет; 4. Формирует массив k из произведений всех элементов каждой строки матрицы H.   *Эталон ответа: г* | Р=4 |
| 14 | **В программе описана матрица**  **A:arrаy[1..n,1..m} of integer;**  **Во время работы программы формируется вектор, в который в порядке возрастания помещаются все элементы матрицы. Определить, какое из приведенных ниже описаний подходит для данного вектора**   1. B: arrаy {1..m+n] of integer; 2. B: arrаy [1..m\*n] of integer; 3. B: arrаy [ n+m] of integer; 4. B: arrаy [1..2\*(m+n)] of integer;   *Эталон ответа: б* | Р=4 |
| 15 | **В каком порядке производится заполнение значениями матрицы размером** 10 x 10 **в следующем фрагменте программы?**  …  **For j:=1 To 10**  **Do for i:= 10 DownTo 1**  **Do A[I,j]:=random(100);**  …   1. По строкам справа налево; 2. По столбцам сверху вниз 3. Правильного ответа нет.; 4. По столбцам снизу вверх   *Эталон ответа: б* | Р=4 |
| 16 | **Что определяет для массива X[1..n, 1..m] следующий алгоритм**  **…**  **S:= 0;**  **For i:= 1 to n do**  **For j:= 1 to m do**  **If X[i, j ] <0 then S:=S+X[i, j ];**  **…**   1. минимальный элемент массива; 2. сумму отрицательных элементов массива; 3. количество отрицательных элементов массива; 4. индекс последнего отрицательного элемента массива.   *Эталон ответа: б* | Р=4 |
| 17 | Оператор присоединения, позволяющий сократить обозначение полей имеет вид:   1. While имя записи do оператор; 2. With имя записи do оператор; 3. Имя переменной=имя поля; 4. Имя записи=record   *Эталон ответа: б* | Р=4 |
| 18 | **Что делает следующий фрагмент программы?**  **…**  **Type**  **Zap=Record**  **X,Y:Real;**  **Z:Integer;**  **S:char;**  **End;**  **…**  **Var A:array[1..30] of Zap;**  **M,i:integer;**  **…**  **M:=A[1].Z;**  **For I:=2 to 30 Do**  **if A[i].Z>M Then M:=A[i].Z;**  …  а) ищет максимальный элемент массива А;  б) ищет минимальный элемент массива А;  в) ищет запись с максимальным значением поля Z;  г) ищет запись с минимальным значением поля Z.  *Эталон ответа: в* | Р=4 |
| 19 | **Какой тип данных не может выступать как базовый тип множества?**   1. Byte; 2. Real; 3. Интервальный; 4. Перечисляемый.   *Эталон ответа: б* | Р=4 |
| 20 | **Что делает следующий фрагмент программы?**  **Type chisla= set of byte;**  **Var z,x,y: chisla;**  **…**  **x:= [0..150];**  **y:= [100..255];**  **z:= y-х**  **…**   1. получено множества чисел из диапазона 100..150; 2. получено множества чисел из диапазона 0..255; 3. получено множества чисел из диапазона 150..255; 4. получено множества чисел из диапазона 0..100.   *Эталон ответа: г* | Р=4 |
| 21 | **Какая операция над множествами показана?**  B  А  B  B  А  А  B   1. Объединение множеств; 2. Пересечение множеств; 3. Разность (В-А) множеств; 4. Разность (А-В) множеств.   *Эталон ответа: б* | Р=4 |
| 22 | Описание множественного типа данных  1. Type <имя\_типа>= set of <базовый\_тип>; 2. Type <имя\_типа>= record of <базовый\_тип>; 3. Type <имя\_типа>:= set of <базовый\_тип>; 4. Type <имя\_типа>: set of <базовый\_тип>.   *Эталон ответа: г* | Р=4 |
| 23 | **Заполните пробелы в следующем предложении Команда … определяет, достигнут ли конец строки.**   1. Eof(файлов\_пер); 2. Eoln(файлов\_пер); 3. Erase(файлов\_пер); 4. Append(файлов\_пер).   *Эталон ответа: б* | Р=4 |
| 24 | **Укажите, как будет записана команда, выполняющая следующие действия. Чтение информации из файла, через файловую переменную del в переменную settr.**   1. Write(del,serrt); 2. Seek (del,serrt); 3. Read(del,serrt); 4. Read(serrt, del).   *Эталон ответа: а* | Р=4 |
| 25 | **Укажите, как будет записана команда, выполняющая следующие действия. В разделе переменных, файловая переменная kl подготавливается для работы с текстовым файлом**   1. Var Kl:File of tekt; 2. Var Kl:text; 3. Var Kl:String; 4. Var Kl:file of char.   *Эталон ответа: б* | Р=4 |

* + 1. Примерный перечень теоретических вопросов

1. Чем определяются ограничения на множество целых чисел?
2. Для представления какой информации используется тип real?
3. Какие две формы записи вещественных чисел используются в языке?
4. Что называется мантиссой числа?
5. Что называется порядком числа?
6. Какие стандартные тригонометрические функции реализованы в языке?
7. Можно ли аргумент стандартной тригонометрической функции задать в градусах?
8. Назовите операции определённые для данных целого типа, но не определенных для данных вещественного типа?
9. Почему запись формул на языке линейная?
10. Как называются слова, имеющие фиксированное начертание, раз и навсегда определенный смысл?
11. Из каких символов может состоять идентификатор?
12. Что такое выражение?
13. Что такое операция?
14. Приведите примеры составных специальных символов.
15. Как называются величины, которые, могут менять своё значение в процессе выполнения программы?
16. Из каких команд состоят линейные алгоритмы?
17. Что называют операторными скобками?
18. Каков будет результат выполнения программы:

Var r,T:real;

Begin

R:=4.869;

T:=trunс(r);

End.

1. Для чего используется указание типа величины?
2. Какое влияние оказывает комментарий на выполнение программы?
3. Для чего нужен комментарий?
4. Каким символом заканчивается текст программы?
5. Какие операции определены над логическими величинами?
6. В чем заключается отличие полной формы оператора условия от сокращенной?
7. Какие служебные слова могут использоваться при записи условного оператора?
8. Какие операторы входят в состав разветвляющихся программ?
9. Как проверить целочисленную переменную на нечетность?
10. Какой тип должно иметь выражение, стоящее в условном операторе между словами if и then
11. Как называется программа, в основе которой лежит структура повторения?
12. Перечислите циклические структуры.
13. Опишите работу цикла с предусловием.
14. Опишите работу цикла с постусловием.
15. Какой тип должен иметь параметр цикла в цикле for?
16. Какие циклы называются вложенными?
17. Какое максимально возможное количество символов может содержать строка?
18. Как при описании строкового типа указывается длина строки?
19. Всегда ли длина строки равна объявленной в описании?
20. Если длина строки не указана явно, то чему она равна по умолчанию?
21. Перечислите основные функции и процедуры для обработки строк?
22. Как происходит сравнение строк?
23. Каков приоритет операций отношения?
24. Является ли тип String структурированным типом данных? Обоснуйте свой ответ.
25. Перечислите типовые операции над строками?
26. Дайте определение подпрограммы?
27. Существуют ли подпрограммы без параметра?
28. Как называются параметры, определяемые в заголовке подпрограммы?
29. Какое количество значений возвращает функция?
30. Каково соответствие между фактическими и формальными параметрами?
31. Какие переменные называются локальными?
32. Может ли в основной программе функция вызываться внутри выражения?
33. Наличие какого оператора необходимо для возвращения значения из функции в вызывающую подпрограмму?
34. Что такое рекурсивные подпрограммы?
35. Дайте определение массива.
36. Где и как определяется общее число элементов массива?
37. Дайте определение индекса?
38. Какие типы данных могут использоваться при описании индекса?
39. Могут ли существовать массивы массивов? Приведите пример
40. Какой массив называется одномерным?
41. Существуют ли ограничения на размерность массива?
42. Дайте определение поля записи.
43. Могут ли записи иметь вложенную структуру?
44. Чем запись отличается от массива?
45. Что называется базовым типом множества?
46. Может ли множество не содержать ни одного элемента?
47. Как работает операция in?
48. Что называется объединением множеств?
49. Какой оператор используется для упрощения доступа к полям записи?
50. Какого типа могут быть компоненты файла?
51. Как определяется длина файла?
52. Можно ли открыть файл одновременно для записи и чтения?
53. В каких случаях стандартная функция eof принимает значение true, а в каких false при чтении файла?
54. Каково значение операции closeFile?
55. Можно ли прочитать файл, открытый для записи? Обоснуйте ответ
56. Какие типы файлов существуют?
57. Как располагаются графические оси координат?
58. Что называют графическими примитивами?
    1. Задания для промежуточной аттестации

Промежуточная аттестация состоит из двух частей :

Тестовое задание (знания)-время выполнения 30 мин;

Практическое задание (умения)-время выполнения 60 мин.

Примерные тестовые задания

**Задание: Выберите правильный ответ и обведите его кружочком.**

Вариант 1

|  |  |  |
| --- | --- | --- |
| 1. | Альтернатива - это:  а) композиция разных действий;  б) вариант;  в) конструкция ветвления;  г) шаг выполнения программы  *Эталон ответа: в* | Р=4 |
| 2. | Итерация-это:  а) шаг выполнения программы;  б) циклическая конструкция алгоритма;  в) язык программирования;  г) функция прерывания.  *Эталон ответа: а* | Р=4 |
| 3. | 1. Как выглядит блок ввода информации в графическом алгоритме? 2. 2 3 4   *Эталон ответа: 4* | Р=4 |
| 4. | Язык программирования образуют три составляющие:  а) алфавит, орфография, диалектика;  б) алфавит, синтаксис, семантика;  в) переменные, процедуры, функции;  г) модули, описание, реализация.  *Эталон ответа: б* | Р=4 |
| 5 | Алфавит языка программирования - это:   1. Фиксированный набор символов, однозначно трактуемых; 2. А я; 3. a..z; 4. набор слов, которые понимает компьютер.   *Эталон ответа: а* | Р=4 |
| 6 | Переменная - это:   1. Объект, способный принимать различные значения; 2. Значения чисел; 3. Меняющееся число; 4. Динамический объект.   *Эталон ответа: а* | Р=4 |
| 7 | Выражение - это :   1. конструкция языка, которая может меняться; 2. текс программы, заключенный в операторные скобки; 3. множество символов, которые является упорядоченными 4. конструкция, задающая правила вычисления значений переменных   *Эталон ответа: г* | Р=4 |
| 8 | Над вещественными величинами определены операции:   1. not, and, or, xor; 2. \*,+,-./; 3. abs( ), sin( ). cos( ),tg( ); 4. trunk( ). Round( ), ord( ).   *Эталон ответа: б* | Р=4 |
| 9 | Как будет выглядеть |х| в записи PASCAL:   1. exp(x) 2. bc(x) 3. abs(x) 4. dec(x)   *Эталон ответа: в* | Р=4 |
| 10 | Для целого числа n, где 100<n<1000 определить символьный вид **третьей** справа цифры в записи числа.   1. Chr(n/100); 2. Chr((n/100)\*10); 3. (n div 1000 mod 10); 4. Chr(( n mod 1000) div 100).   *Эталон ответа:г* | Р=4 |
| 11 | Как правильно записать выражение на языке PASCAL   1. y:=sqrt(exp(l)\*(sqr(x)-l)) 2. Sqr(exp(x)\*ln(x)-1) 3. y:=sqrt(exp(l)\*sqr(x)-l) 4. y:=sqr(exp(l)\*(sqr(x)-l))   *Эталон ответа:а* | Р=4 |
| 12 | Какое служебное слово используют для описания действительного типа данных.   1. word 2. real 3. byte 4. integer   *Эталон ответа:б* | Р=4 |
| 13 | Раздел описания глобальных переменных начинается с служебного слова:   1. LABEL 2. VAR 3. USES 4. TYPE ,   *Эталон ответа:б* | Р=4 |
| 14 | Оператор присваивания имеет вид:   * + - 1. =       2. Begin….end       3. :=       4. Floattostr(s)   *Эталон ответа:в* | Р=4 |
| 15 | Цикл с постусловием запишется в виде:   1. While <Логическое выражение> do <оператор>; 2. For i:=l to n do <оператор>; 3. Repeat <последовательность операторов> until <логическое выражение>; 4. Case k of <Последовательность операторов>.   *Эталон ответа:в* | Р=4 |
| 16 | Вычислить y=e+sin (х) при 0,1 > х ≥-12   1. if (x<0.1) or (х>=-12) then y=exp(l)+sin(x); 2. for х:=0.1 to 12 do y:=exp(x)+sin(x); 3. if (x <0.1) and (x >=-12) then y:=exp(l)+sin(x); 4. if (x <0.1) and (x =-12) then y:=exp(x)+sin(x).   *Эталон ответа:в* | Р=4 |
| 17 | Логическое "ИЛИ":   1. or 2. end 3. else 4. and   *Эталон ответа: а* | Р=4 |
| 18 | Рекурсия - это:   1. Повторение вычисления функции или процедуры внутри себя: 2. Оператор; 3. Цикл; 4. Метод определения функции или процедуры   *Эталон ответа: а* | Р=4 |
| 19 | Обращение к процедуре в программе имеет вид:   1. Оператор процедуры; 2. Имя процедуры(список фактических параметров); 3. Имя процедуры: тип значения; 4. Имя процедуры(список формальных параметров)   *Эталон ответа: б* | Р=4 |
| 20 | Глобальные переменные действуют   1. Во всех процедурах; 2. Во всех функциях; 3. Во всех модулях; 4. Во всей программе.   *Эталон ответа: г* | Р=4 |
| 21 | Процедуры и функции - это:   1. операторы; 2. подпрограммы; 3. имена; 4. переменные.   *Эталон ответа: б* | Р=4 |
| 22 | Delphi относят к системам ...программирования.   1. визуального; 2. наглядного; 3. системного; 4. быстрого   *Эталон ответа: а* | Р=4 |
| 23 | Главный файл проекта, представляющий собой основной модуль программы, имеет расширение:   1. .pas; 2. .dpr; 3. .res; 4. .dsk.   *Эталон ответа: б* | Р=4 |
| 24 | Функция, позволяющая преобразовать значение целочисленного выражения в строку:   1. IntToStr(s:Integer):String; 2. StrToInt(s:String):Integer; 3. SrtToFloat(S:String):Real; 4. FloatToStr(a:Real): String.   *Эталон ответа: а* | Р=4 |
| 25 | Запретить изменять размеры окна позволяет свойство:   1. BorderStyle; 2. Visible; 3. Active; 4. Position.   *Эталон ответа: а* | Р=4 |
| 26 | Для определения комбинации клавиш быстрого доступа необходимо в заголовке перед соответствующим символом поставить:   1. % 2. $ 3. & 4. №   *Эталон ответа: в* | Р=4 |
| 27 | Свойство алгоритма означающее, что переход к следующему шагу возможен лишь после завершения предыдущего шага:   1. результативность 2. дискретность 3. определенность 4. массовость.   *Эталон ответа: б* | Р=4 |
| 28 | К какому типу данных относится константа '4'   1. Char 2. Boolean 3. Real 4. Integer   *Эталон ответа: а* | Р=4 |
| 29 | Интервальный тип данных описывается:   1. Имя\_типа:=1,2,….,3,4… 2. Имя-типа=max..min; 3. Имя-типа:=min…max; 4. Имя-типа:=Min..max;   *Эталон ответа: в* | Р=4 |
| 30 | Переменная логического типа может принимать значения   1. Истина и ложь 2. False, tryu 3. False, true 4. Fales,true   *Эталон ответа: в* | Р=4 |
| 31 | Свойство Caption задаёт   1. Текст заголовка 2. Название компонента 3. Имя компонента 4. Текст компонента   *Эталон ответа: а* | Р=4 |
| 32 | В результате выполнения функции на экране появляется диалоговое окно, в поле которого пользователь может ввести строку символов.   1. ShowMessage(S); 2. MessageDlg(s,t,b,h); 3. InputBox(Заголовок, Подсказка, Значение); 4. InputBox(f);   *Эталон ответа: в* | Р=4 |
| 33 | Процедура, которая выводит окно с сообщением и командной кнопкой OK.   1. InputBox(Заголовок, Подсказка, Значение); 2. ShowMessage(S); 3. MessageDlg(s,t,b,h); 4. InputBox(f);   *Эталон ответа: б* | Р=4 |
| 34 | Передайте смысл заголовка процедуры  Procedure TForm1.Button3Click(Sender:TObject);   1. Три раза щелкнуть на кнопке; 2. Поместить на форму три кнопки; 3. Один раз щелкнуть на кнопке ; 4. Один раз щелкнуть на форме.   *Эталон ответа: в* | Р=4 |
| 35 | Свойство позволяющее сделать объект скрытым или видимым.   1. Transparent 2. Visible 3. Enabled 4. Alignment   *Эталон ответа :б* | Р=4 |
| 36 | Свойство, задающее способ выравнивания текста внутри поля.   1. Transparent 2. Visible 3. Enabled 4. Alignment   *Эталон ответа: г* | Р=4 |
| 37 | К какому типу данных относится число -543.12   1. Real 2. Integer 3. Char 4. Boolean   *Эталон ответа: а* | Р=4 |
| 38 | Какое действие обязательно должно быть в теле функции   1. Вызов функции; 2. Имя\_функции=результат\_функции; 3. Имя\_функции:=значение; 4. Function имя.   *Эталон ответа: в* | Р=4 |
| 39 | **Сколько раз будет выполняться цикл For i:=4 to 8 do…;**   * 1. 5 раз;   2. 6 раз;   3. 8 раз;   4. 4 раза.   *Эталон ответа: а* | Р=4 |
| 40 | **Константы ─ это:**   1. величины, которые могут менять свое значение в процессе выполнения программы 2. величины, которые не могут менять своего значения в процессе выполнения программы 3. обозначают строки программы, на которые передается управление во время выполнение программы 4. Верного ответа нет   *Эталон ответа: б* | Р=4 |

Вариант 2

|  |  |  |
| --- | --- | --- |
| 1. | В каком ответе верно объявлен одномерный массив?   1. VAR m:integer; 2. VAR m[i]:integer; 3. VAR m:array[1..10] of real; 4. VAR m[i] of real   *Эталон ответа: в* | Р=4 |
| 2. | Чему равно s после отработки цикла, если массив имеет вид: 0 7 6 6 2 7 1 2 8 5  s:=0;  for i:=1 to 5 do s:=s+m[i];   1. s=21; 2. s=43; 3. s=19; 4. s=28.   *Эталон ответа: а* | Р=4 |
| 3. | В массиве А для обращения к его элементу необходима следующая запись:   1. A,I; 2. I; 3. A[i]; 4. A{i}.   *Эталон ответа: в* | Р=4 |
| 4. | Цикл с предусловием запишется в виде:   1. While <Логическое выражение> do <оператор>; 2. For i:=l to n do <оператор>; 3. Repeat <последовательность операторов> until <логическое выражение>; 4. Case k of <Последовательность операторов>.   *Эталон ответа: а* | Р=4 |
| 5 | Сколько индексов может содержать одномерный массив?   1. 2 2. 1 3. 10 4. хоть сколько   *Эталон ответа: б* | Р=4 |
| 6 | Множества – это :   1. cписок элементов, заключенный в круглые скобки, вида: <имя поля>:<значение>; 2. неупорядоченная совокупность отличных друг от друга однотипных элементов; 3. последовательность с фиксированным числом однотипных элементов, отличающихся индексами; 4. совокупность отличных друг от друга элементов разных типов.   *Эталон ответа: б* | Р=4 |
| 7 | Каков будет результат выполнения программы  Var S:byte;  Begin  S:=length(‘каникулы’);  End.   1. S=0 2. S=1 3. S=8 4. S=true   *Эталон ответа: в* | Р=4 |
| 8 | Каков будет результат выполнения программы  Type digits=set of 0..9;  Var d1, d2, d3:digits;  Begin  d1:=[1,3,5];  d2:=[0,4,5];  d3:=d1\*d2;  end.   1. d3=[0,1,3,4,5] 2. d3=[0,1,3,4] 3. d3=[5] 4. d3=[1,3,5,0,4,5]   *Эталон ответа: в* | Р=4 |
| 9 | В типе String количество символов одной строки не должно превышать;   1. 256 2. 255 3. 1024 4. 2400   *Эталон ответа: а* | Р=4 |
| 10 | Каково будет значение переменной s1  Var s1,s2:string;  Begin  S1:=’информатика’;  Delete(s1,3,4);  End.   1. Инатика; 2. Форма; 3. Инф; 4. Инфо.   *Эталон ответа: а* | Р=4 |
| 11 | Операция, используемая для определения наличия эл. во множестве   1. on; 2. of; 3. in; 4. up.   *Эталон ответа: в* | Р=4 |
| 12 | Массив – это :   1. запись множества переменных разного типа; 2. неупорядоченная совокупность отличных друг от друга однотипных элементов; 3. последовательность, состоящая из фиксированного числа однотипных элементов; 4. тип одномерных величин.   *Эталон ответа: в* | Р=4 |
| 13 | Сколько элементов содержит одномерный массив m[10]?   1. i 2. 10 3. M[10]-i 4. 9   *Эталон ответа: б* | Р=4 |
| 14 | Над файловыми величинами определены операции:   1. not, and, or и стандартные; 2. <, >, +, odd(), abs(),и стандартные; 3. eof(), eoln(), seek(), rewrite() и стандартные; 4. eof(), eoln(), seek(), rewrite().   *Эталон ответа: а* | Р=4 |
| 15 | Открыть файл в режиме до записи:   1. Append(); 2. Reset(); 3. Rewrite(); 4. Write().   *Эталон ответа: а* | Р=4 |
| 16 | В программе описан двумерный массив : a:array[1..n,1..m] of integer. Во время работы программы формируется одномерный массив, в котором в порядке возрастания помещаются все элементы массива. Определить, какое из приведенных ниже описаний подходит под данный массив.   1. B:array[1..n\*m] of integer; 2. B:array[1..n+m] of integer; 3. B:array[n+m] of integer; 4. B:array[1..2\*(n+m)] of integer;   *Эталон ответа: а* | Р=4 |
| 17 | Запись – это:   1. константное значение; 2. последовательность, состоящая из фиксированного числа однотипных элементов; 3. последовательность однотипных элементов, отличающихся индексами; 4. последовательность, состоящая из фиксированного числа величин, называемых полями.   *Эталон ответа: г* | Р=4 |
| 18 | Операция, обнаруживающая состояние конца файла:   1. OEF(); 2. OLE(); 3. EOF(); 4. EOLN().   *Эталон ответа: в* | Р=4 |
| 19 | Объединение полей, методов и свойств объекта называется:   1. Инкапсуляция; 2. Наследование; 3. Полиморфизм; 4. Рекурсивность.   *Эталон ответа: а* | Р=4 |
| 20 | **Условный оператор применяется для программирования:**   1. Сложных алгоритмов 2. Линейных алгоритмов 3. Ветвящихся алгоритмов 4. Циклических алгоритмов   *Эталон ответа: в* | Р=4 |
| 21 | Понятие, сочетающее в себе совокупность данных и действий над ними:   1. Объект; 2. Свойство; 3. Метод; 4. Класс.   *Эталон ответа: а* | Р=4 |
| 22 | Оператор присоединения, позволяющий сократить обозначение полей имеет вид:   1. While имя записи do оператор; 2. With имя записи do оператор; 3. Имя переменной. имя поля; 4. Имя записи=record   *Эталон ответа: б* | Р=4 |
| 23 | **Какой из операторов цикла имеет 2 вида записи**   * 1. For…   2. While…   3. Repeat…   4. Верного ответа нет   *Эталон ответа: а* | Р=4 |
| 24 | Что называют конструктором множества?   1. Набор элементов множества; 2. Базовый тип; 3. Перечисление элементов множества через запятую, заключенных в квадратные скобки; 4. Все множества в квадратных скобках.   *Эталон ответа: в* | Р=4 |
| 25 | Что называют произведением множеств?   1. Множество, состоящее из всех элементов принадлежащих хотя бы одному из множеств; 2. Множество, состоящее из элементов принадлежащих одновременно всем исходным множествам; 3. Множество, в которое входят все элементы первого множества, не входящие во второе множество. 4. Множество, которому принадлежат те и только те элементы, которые принадлежат хотя бы одному из множеств.   *Эталон ответа: б* | Р=4 |
| 26 | Процедура установки соответствия между файловой переменной и файлом на диске.   1. Erase(fm); 2. Reset(Fm); 3. AssignFile(Fm,’путь к файлу’); 4. Rewrite(fm).   *Эталон ответа: в* | Р=4 |
| 27 | Процедура удаления неоткрытого файла с диска.   1. Erase(fm); 2. Reset(Fm); 3. AssignFile(Fm,’путь к файлу’); 4. Rewrite(fm).   *Эталон ответа: а* | Р=4 |
| 28 | Процедура, открывающая файл для чтения.   1. Erase(fm); 2. Reset(Fm); 3. AssignFile(Fm,’путь к файлу’); 4. Rewrite(fm).   *Эталон ответа: а* | Р=4 |
| 29 | Функция, определяющая фактическую длину строки.   1. Copy(a, b, c); 2. Delete(a, b, c); 3. Length(st); 4. Pos(st1, st2).   *Эталон ответа: в* | Р=4 |
| 30 | Удаление из строки определенного числа символов, начиная с указанной позиции.   1. Copy(a, b, c); 2. Delete(a, b, c); 3. Length(st); 4. Pos(st1, st2)   *Эталон ответа: б* | Р=4 |
| 31 | Три понятия характеризующие, объектно-ориентированное программирование.   1. Конструкторы и деструкторы, события; 2. Полиморфизм, инкапсуляция и наследование; 3. Классы, объекты и методы; 4. Имя, состояние и метод.   *Эталон ответа: б* | Р=4 |
| 32 | Функция, осуществляющая поиск определенного фрагмента в строке.   1. Copy(a, b, c); 2. Delete(a, b, c); 3. Length(st); 4. Pos(st1, st2)   *Эталон ответа: г* | Р=4 |
| 33 | Раздел описания типов начинается со служебного слова:   1. LABEL 2. VAR 3. USES 4. TYPE   *Эталон ответа:г* | Р=4 |
| 34 | **Цикл REPEAT называется**   1. Цикл с предусловием 2. Цикл с параметром 3. Цикл с постусловием 4. Ветвлением   *Эталон ответа: а* | Р=4 |
| 35 | **Каков будет результат выполнения фрагмента программы?**  **Var S1:string;**  **…**  **S1:=Copy('крокодил',4,3);**  …   1. крок; 2. одил; 3. кродил; 4. код.   *Эталон ответа: г* | Р=4 |
| 36 | **Чему будет равно значение переменной М в результате выполнения серии операторов**  **N:=5;**  **Y:= 9 mod 4;**  **If Y>0 then M:=N/5+3\*Y\*N**   1. 31 2. 16 3. 0 4. 5   *Эталон ответа: б* | Р=4 |
| 37 | **Остаток от целочисленного деления можно выразить следующей функцией:**   1. A mod B 2. A div B 3. Abs (A)\*B 4. Верного ответа нет   *Эталон ответа: а* | Р=4 |
| 38 | **Тип переменных REAL это:**   1. целочисленный тип 2. логический тип 3. натуральный тип 4. верного ответа нет   *Эталон ответа: г* | Р=4 |
| 39 | **Оператор CASE служит для создания:**   1. Линейных алгоритмов 2. Ветвящихся алгоритмов 3. Циклических алгоритмов 4. Верного ответа нет   *Эталон ответа: а* | Р=4 |
| 40 | **Переменные – это:**   1. величины, которые могут менять свое значение в процессе выполнения программы 2. величины, которые не могут менять своего значения в процессе выполнения программы 3. обозначают строки программы, на которые передается управление во время выполнение программы 4. Верного ответа нет   *Эталон ответа: а* | Р=4 |

**Методика оценивания качества выполнения заданий**

**(В.П. Беспалько)**

Количественным критерием оценки правильности выполнения тестовых заданий служит коэффициент Ка, представляющий собой отношение количества правильно выполненных обучающимися существенных операций (А) к общему числу существенных операций теста (Р) Ка = А/Р

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ка | 1,0-0,9 | 0,89-0,8 | 0,79-0,7 | < 0,7 |
| отметка | 5 | 4 | 3 | 2 |

Примерные практические задания

**Задание:**

* Осуществите разработку кода программного продукта на основе готовой спецификации;
* Выполните отладку программного продукта с использованием специализированных программных средств;
* Выполните тестирование программного продукта.

Вариант 1

(выходные данные для проверки а=16**)**

b=4

b:=b+1

a:=a\*2

a:=1

b:=0

да

нет

Вариант 2

(выходные данные для проверки k=3)

K<>t

k:=81

t:=48

k>t

k:=k-t

t:=t-k

нет

да

да

нет

K<>t

k:=81

t:=48

k>t

k:=k-t

t:=t-k

нет

да

да

нет

Вариант 3

(выходные данные для проверки n=160)

m>6

n:=10

m:=12

m:=m-2

n:=n\*2

да

нет

Вариант 4

(выходные данные для проверки c=2)

b=n

c:=b-c

b:=b+1

b:=1

c:=0

n:=5

да

нет

Оценивание качества программного продукта производится в соответствии с критериями оценки программного продукта согласно показателям качества по ГОСТ 28195-89

**3. Рекомендуемая литература и иные источники**

1. Семакин И.Г., Шестаков А.П. Основы алгоритмизации и программирования: Учебник для сред. проф. образования / И.Г. Семакин, А.П. Шестаков. - М.: Издательский центр "Академия", 2012. - 400 с. (Допущено Министерством образования и науки Российской Федерации Э
2. Немцова Е. И., Голова С. Ю., Абрамова И. В. Программирование на языке высокого уровня. Программирование на языке Objekt Pascal: учебное пособие—М., ИД «Форум»: ИНФРА-М, 2009.-- 496 с.: ил.

**Дополнительные источники:**

1. Архангельский А. Я. Программирование в Delphi для Windows. Turbo Delphi. М.: Бином, 2009.-350с.
2. *Дж. Рубенкинг.* Язык программирования Delphi для «чайников». Введение в Borland Delphi 2006 = Delphi for Dummies — М.:Диалектика, 2007. — 336 с.
3. *Хавьер Пашеку.* Программирование в Borland Delphi 2006 для профессионалов = Delphi for .NET Developer’s Guide — М.:Вильямс, 2006. — 944 с.
4. Культин Н.Б. Программирование в Turbo Pascal и Delphi. - СПб.: BHV - Санкт-Петербург, 1998. - 240 с.