

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Коротков Сергей Леонидович  
Должность: Директор филиала СамГУПС в г. Ижевске  
Дата подписания: 03.08.2023 09:10:26  
Уникальный программный ключ:  
d3cff7ec2252b3b19e5caaa8cefa396a11af1dc5

Приложение  
к ППССЗ по специальности 09.02.07  
Информационные системы и программирование

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ УЧЕБНОЙ ПРАКТИКЕ

### УП.01.01 УЧЕБНАЯ ПРАКТИКА

### ПМ.02 РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СЕТЕЙ

### 09.02.07 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ПРОГРАММИРОВАНИЕ

базовый уровень подготовки

Год начала подготовки – 2023

## Оглавление

1. Паспорт фонда оценочных средств.....	4
2. Результаты освоения программы учебной практики, подлежащие проверке...	5
3. Типовые задания для оценки освоения программы учебной практики.....	8
3.1 Основные источники.....	17
4. Контрольно-оценочные материалы для итоговой аттестации .....	18

## **1. Паспорт фонда оценочных средств**

Фонд оценочных средств предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной практики профессионального модуля ПМ.01. Разработка модулей программного обеспечения для компьютерных сетей.

ФОС включают контрольные материалы для проведения текущего контроля и промежуточной аттестации в форме дифференцированного зачёта.

ФОС разработан на основании положений:

программы подготовки специалистов среднего звена по специальности СПО 09.02.07 Информационные системы и программирование;

программы учебного модуля;

учебного плана по специальности СПО 09.02.07 Информационные системы и программирование;

положения «О фонде оценочных средств для проведения текущего контроля успеваемости промежуточной и итоговой аттестации студентов и обучающихся филиала СамГУПС в г. Алатыре».

## 2. Результаты освоения программы учебной практики, подлежащие проверке

### 2.1 Перечень умений, знаний, общих компетенций

С целью овладения указанным видом профессиональной деятельности и соответствующими профессиональными компетенциями обучающийся в ходе освоения рабочей программы учебной практики должен:

#### 1.1.1.Перечень общих компетенций

ОК 1.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;
ОК 2.	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;
ОК 3	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях;
ОК 4	Эффективно взаимодействовать и работать в коллективе и команде;
ОК 5	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста;
ОК 6	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения;
ОК 7	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях;
ОК 8	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности;
ОК 9	Пользоваться профессиональной документацией на государственном и иностранном языках.»;

#### 1.1.2.Перечень профессиональных компетенций

ПК 1.1	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
ПК 1.2	Разрабатывать программные модули в соответствии с техническим заданием
ПК 1.3	Выполнять отладку программных модулей с использованием специализированных программных средств
ПК 1.4	Выполнять тестирование программных модулей

1.1.3. В результате освоения профессионального модуля студент должен:

Иметь практический опыт	В разработке кода программного продукта на основе готовой спецификации на уровне модуля; использовании инструментальных средств на этапе отладки программного продукта; проведении тестирования программного модуля по определенному сценарию; использовании инструментальных средств на этапе отладки программного продукта; разработке мобильных приложений
уметь	осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; осуществлять разработку кода программного модуля на современных языках программирования; уметь выполнять оптимизацию и рефакторинг программного кода; оформлять документацию на программные средства
знать	основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования; способы оптимизации и приемы рефакторинга; основные принципы отладки и тестирования программных продуктов

### 3. Типовые задания для оценки освоения программы учебной практики

Работа содержит задания по разработке программного обеспечения с использованием инструментальных средств. Все документы должны быть выполнены максимально точно по представленному образцу.

Результаты выполнения задания оформляются в виде отдельных файлов соответствующих форматов и сохраняются на ПК. Для проверки и оценки результаты выполнения экзаменационного задания предоставляются в электронном виде.

В процессе выполнения задания вы можете воспользоваться методическими пособиями, предоставленной учебной литературой и информацией сети Интернет.

#### Тестирование

#### Системное программирование

#### Раздел 1. Введение в системное программирование

1. Операционная система:
  - a) **система программ, которая обеспечивает совместную работу всех устройств компьютера по обработке информации**
  - b) система математических операций для решения отдельных задач
  - c) система планового ремонта и технического обслуживания компьютерной техники
2. Программное обеспечение (ПО) – это:
  - a) **совокупность программ, позволяющих организовать решение задач на компьютере**
  - b) возможность обновления программ за счет бюджетных средств
  - c) список имеющихся в кабинете программ, утвержден администрацией школы
3. Загрузка операционной системы – это:
  - a) запуск специальной программы, содержащей математические операции над числами
  - b) **загрузка комплекса программ, которые управляют работой компьютера и организуют диалог пользователя с компьютером**
  - c) вложение дискеты в дисковод
4. Система программирования – это:
  - a) комплекс любимых программ программиста
  - b) **комплекс программ, облегчающий работу программиста**
  - c) комплекс программ, обучающих начальным шагам программиста
5. Прикладное программное обеспечение – это:
  - a) справочное приложение к программам
  - b) **текстовый и графический редакторы, обучающие и тестирующие программы, игры**
  - c) набор игровых программ
6. Прикладное программное обеспечение:
  - a) программы для обеспечения работы других программ
  - b) **программы для решения конкретных задач обработки информации**
  - c) программы, обеспечивающие качество работы печатающих устройств
7. Операционные системы:

- a) **DOS, Windows, Unix**
  - b) Word, Excel, Power Point
  - c) (состав отделения больницы): зав. отделением, 2 хирурга, 4 мед. Сестры
8. Системное программное обеспечение:
- a) **программы для организации совместной работы устройств компьютера как единой системы**
  - b) программы для организации удобной системы размещения программ на диске
  - c) набор программ для работы устройства системного блока компьютера
9. Сервисные (обслуживающие) программы:
- a) программы сервисных организаций по бухгалтерскому учету
  - b) программы обслуживающих организаций по ведению делопроизводства
  - c) **системные оболочки, утилиты, драйвера устройств, антивирусные и сетевые программы**
10. Системные оболочки – это:
- a) специальная кассета для удобного размещения дискет с операционной системой
  - b) **специальная программа, упрощающая диалог пользователь – компьютер, выполняет команды операционной системы**
  - c) система приемов и способов работы конкретной программы при загрузке программ и завершении работы
11. Использование одного имени для задания общих для класса действий, что означает способность объектов выбирать внутренний метод, исходя из типа данных, определяет свойство ООП
- a) Полиморфизм
  - b) Управление событиями
  - c) **Инкапсуляция**
  - d) Наследование
12. Понятие «инкапсуляция» относится к
- a) Технологии модульного программирования
  - b) **Технологии объектно – ориентированного программирования**
  - c) Технологии императивного программирования
  - d) Технологии модульного программирования
- 13.Свойство ООП, которое может быть смоделировано с помощью таксономической классификационной схемы (иерархии) называется
- a) Инкапсуляция
  - b) Управление событиями
  - c) Полиморфизм
  - d) **Наследование**
- 14.Понятие класса в ООП включает в себя
- a) **Поля и методы класса**
  - b) Процедуры и функции обработки
  - c) Поля и функции обработки
  - d) Поля и процедуры обработки
- 15.Назначение конструктора объекта
- a) **Только выделяет память под объект**
  - b) Выделяет память и задает начальное значение полям

- c) Задаёт начальное значение полям
  - d) Выделяет память, задаёт начальное значение полям, выполняет любые проверки, заданные программистом
16. Как описывается конструктор объекта
- a) procedure create;
  - b) constructor create;
  - c) function create;
  - d) function constructor;**
17. Как описывается деструктор объекта
- a) procedure free;
  - b) destructor free;**
  - c) free;
  - d) function free;
18. Понятия объекта в ООП - это
- a) представитель класса
  - b) конкретные данные, заданные в классе.**
  - c) компонент панели инструментов
  - d) встроенный объект Delphi
19. Моделями типа «чёрный ящик» являются
- a) Модели мышления
  - b) Модели, описывающие зависимость параметров состояния объекта от входных параметров
  - c) Модели, описывающие входные и выходные параметры объекта без учёта внутренней структуры объекта**
  - d) Модели «аварийного» ящика на самолетах
20. Моделями типа «белый ящик» являются
- a) Модели мышления
  - b) Модели, описывающие зависимость параметров состояния объекта от входных параметров**
  - c) Модели, описывающие входные и выходные параметры объекта с учётом внутренней структуры объекта
  - d) Модели, описывающие выходные данные в программе
21. Программа «драйвер» служит для
- a) запуска программы на выполнение
  - b) имитации запуска программы на выполнение
  - c) проверки правильности работы программы**
  - d) передачи параметров в процедуры и функции
22. Программа «заглушка» служит для
- a) запуска программы на выполнение
  - b) имитации запуска другой программы на выполнение**
  - c) проверки правильности работы программы
  - d) имитации передачи параметров в другой модуль
23. Какие методы сборки программы существуют
- a) монолитная



- b) **пошаговая**
- c) **одновременная**
- d) постепенная

24. Какой метод тестирования программы учитывает закон распределения входных данных

- a) детерминированное тестирование
- b) функциональное тестирование
- c) стохастическое тестирование
- d) **логическое тестирование**

25. Программирование сверху вниз – это

- a) Процесс, при котором от начального предположения осуществляется движение по направлению к лучшим решениям
- b) **Процесс пошагового разбиения алгоритма на все более мелкие части с целью получения таких элементов, для которых можно написать конкретные команды**
- c) Метод сведения трудной задачи к последовательности более простых
- d) Исследование древовидной модели пространства решений и ориентация на поиск оптимального решения

26. Загрузочный модуль программы – результат работы

- a) Грамматики
- b) **Транслятора**
- c) Интерпретатора
- d) Редактора связей (компоновщика)

27. Интегрированная система программирования включает компонент для перевода исходного текста программы в машинный код, который называется

- a) строителем кода
- b) **компилятор**
- c) переводчиком
- d) преобразователем

28. Результатом компиляции программы на языке высокого уровня является

- a) Командный файл
- b) **Объектный файл**
- c) Исходный текст программы на языке высокого уровня
- d) Дисплейный файл

## **Раздел 2. Основы Ассемблер**

1. Установите порядок, в котором осуществляется перевод произвольного двоичного числа в системе счисления с основанием  $q=2^n$

Укажите порядок следования всех 3 вариантов ответа: 1. рассмотреть каждую группу как  $n$ -разрядное двоичное число и записать ее соответствующей цифрой в системе счисления с основанием  $q = 2^n$ ; 2. двоичное число разбить слева и справа (целую и дробную части) на группы по цифр в каждой; 3. если в последних правой и левой группах окажется меньше разрядов, то их надо дополнить справа и слева нулями до нужного числа разрядов

- a) **3 1 2**
- b) 2 1 3

- c) 1 2 3  
d) 3 2 1
2. По сколько цифр в группе нужно разбивать двоичное число при переводе его в четверичную систему счисления?  
a) 6  
**b) 2**  
c) 3  
d) 4
3. Переведите двоичное число 100011111011 в шестнадцатеричную систему счисления:  
a) 5fa  
**b) 8fb**  
c) 8fc  
d) 8fa
4. Чему будет равен результат сложение двоичных чисел 1001 и 1000  
**a) 10001**  
b) 11001  
c) 10011  
d) 10101
5. Выполните деление в двоичной системе счисления  $1110:10=$   
a) 101  
b) 011  
c) 110  
**d) 111**
6. Переведите двоичное число 1010101 в восьмеричную систему счисления  
**a) 125**  
b) 124  
c) 126  
d) 123
7. Установите порядок, в котором осуществляется перевод целого числа в системе счисления с основанием  $q=2^n$ . Укажите порядок следования всех 3 вариантов ответа:  
1. данное двоичное число разбить справа налево на группы по  $n$  цифр в каждой; 2. рассмотреть каждую группу как  $n$ -разрядное двоичное число и записать ее соответствующей цифрой в системе счисления с основанием  $q = 2^n$ ; 3. если в последней левой группе окажется меньше разрядов, то ее надо дополнить слева нулями до нужного числа разрядов  
**a) 1 3 2**  
b) 1 2 3  
c) 3 2 1  
d) 2 1 3
8. Переведите шестнадцатеричное число FACC в двоичную систему счисления  
a) 1111001011011001  
**b) 1111101011001100**  
c) 1011010111111100  
d) 1101100111011010

9. Выполните вычисление в двоичной системе счисления  $110101-101=$
- 110100
  - 110010
  - 110000**
  - 110001
10. В саду 100 фруктовых деревьев - 14 яблонь и 42 груши. В какой системе счисления посчитаны деревья?
- В шестеричной**
  - В шестнадцатеричной
  - В двоичной
  - В восьмеричной
11. Умножьте в двоичной системе счисления  $1111*11=$
- 1111111
  - 101111
  - 1010101
  - 101101**
12. Установите соответствие. Укажите соответствие для всех 4 вариантов ответа: 1) 4  
2) 5 3) 6 4) 7
- 4=101 5=110 6=100 7=111
  - 5=101 6=110 4=100 7=111**
  - 6=101 5=110 4=100 7=111
  - 4=101 7=110 6=100 5=111
13. Выполните умножение в системе счисления  $10101*101=$
- 110101011
  - 1101001**
  - 111001001
  - 1010101
14. Установите соответствие. Укажите соответствие для всех 4 вариантов ответа:
- A=1010 B=1111 F=1110 E=1011
  - A=1010 F=1111 E=1110 B=1011**
  - A=1010 C=1111 F=1110 E=1011
  - A=1010 B=1111 F=1110 C=1011
15. В какой системе счисления будет верным равенство  $7+8=16$ ?
- 12
  - 6
  - 9**
  - 16
16.  $FFFF+1=$
- 1FFF
  - 10000**
  - FFFE
  - FFF1
17. Выполните вычитание в двоичной системе счисления  $1111-101$
- 1010**
  - 1110

- c) 1100
  - d) 1001
18. Результат деления в двоичной системе счисления числа 110 на 11 равен
- a) 11
  - b) 10**
  - c) 01
  - d) 101
19. Перевести смешанное двоичное 11110000,101 число в восьмеричную систему счисления
- a) 610,4
  - b) 740,5**
  - c) 170,5
  - d) 1E0,A
20. Переведите восьмеричное число 0,25 в двоичную систему счисления
- a) 0,010101**
  - b) 0,101010
  - c) 0,110111
  - d) 0,101011
21. Алгоритм-это:
- a) Указание на выполнение действий,
  - b) Система правил, описывающая последовательность действий, которые необходимо выполнить для решения задачи,**
  - c) Процесс выполнения вычислений, приводящих к решению задачи
22. Свойство алгоритма – дискретность, выражает, что:
- a) Команды должны следовать последовательно друг за другом,
  - b) Каждая команда должна быть описана в расчете на конкретного исполнителя,
  - c) Разбиение алгоритма на конечное число команд**
23. Формальное исполнение алгоритма – это:
- a) Исполнение алгоритма конкретным исполнителем с полной записью его рассуждений,
  - b) Разбиение алгоритма на конкретное число команд и пошаговое их исполнение,
  - c) Исполнение алгоритма не требует рассуждений, а осуществляется исполнителем автоматически**
  - d) Исполнение алгоритма осуществляется исполнителем на уровне его знаний
24. Самое важное свойство алгоритма:
- a) Визуальность,
  - b) Массовость,**
  - c) Дискретность,
  - d) Аудиальность,
25. Какой алгоритм называется линейным:
- a) Выполнение операций зависит от условия,
  - b) Операции выполняются друг за другом,**
  - c) Одни и те же операции выполняются многократно
  - d) Присутствие всех возможных операций в одном алгоритме
26. Графическое задание алгоритма – это:

- a) **Способ представления алгоритма с помощью геометрических фигур,**
  - b) Представление алгоритма в форме таблиц и расчетных формул,
  - c) Система обозначений и правил для единообразной и точной записи алгоритмов и их исполнения
27. В расчете на кого должен строиться алгоритм:
- a) В расчете на ЭВМ,
  - b) В расчете на умственные способности товарища,
  - c) **В расчете на конкретного исполнителя**
28. Какое из перечисленных свойств относится к свойствам алгоритма:
- a) Визуальность,
  - b) Совокупность,
  - c) Аудиальность,
  - d) **Понятность**
29. Псевдокоманда DB означает
- a) **определить байт;**
  - b) определить слово (2 байта);
  - c) определить двойное слово (4 байта);
  - d) определить 6 байт
  - e) определить учетверенное слово (8 байт);
30. Псевдокоманда DW означает
- a) определить байт;
  - b) **определить слово (2 байта);**
  - c) определить двойное слово (4 байта);
  - d) определить 6 байт
  - e) определить учетверенное слово (8 байт);
31. Псевдокоманда DD означает
- a) определить байт;
  - b) определить слово (2 байта);
  - c) **определить двойное слово (4 байта);**
  - d) определить 6 байт
  - e) определить учетверенное слово (8 байт);
32. К сегментным регистрам относят
- a) **CS, DS, SS и ES.**
  - b) AX, BX, CX и DX
  - c) SP и BP.
  - d) SI и DI
33. К индексным регистрам относят
- a) CS, DS, SS и ES.
  - b) AX, BX, CX и DX
  - c) SP и BP.
  - d) **SI и DI**
34. Устанавливает бит результата в 1, если оба бита, бит источника и бит приемника установлены в 1.
- a) AND
  - b) **OR**

- c) XOR (НЕ ИЛИ)
  - d) NOT
35. устанавливает бит результата в 1, если бит источника отличается от бита приемника.
- a) **AND**
  - b) OR
  - c) XOR
  - d) NOT
36. Запись регистра в память осуществляется при помощи команды
- a) **MOV**
  - b) XCHG
  - c) XLAT
37. Порядковый номер элемента в массиве, который всегда начинается с нуля
- a) **Индекс**
  - b) Команда
  - c) Псевдокоманда
  - d) Макрокоманда
38. Позволяет сохранять длину исходного текста программы
- a) Индекс
  - b) Команда
  - c) Псевдокоманда
  - d) **Макрокоманда**

## Тестирование

### Разработка программных модулей

#### Раздел 3. Объектно-ориентированное программирование

1. Delphi – это...
  - 1) Язык программирования низкого уровня
  - 2) **Система объектно-ориентированного визуального программирования**
  - 3) Программа обработки видео
  - 4) Программа для перевода кода программы на язык машинных кодов.
2. RAD – это...
  - 1) Программа записи видео
  - 2) Средство модульного программирования
  - 3) **Среда быстрой разработки приложений**
3. Язык программирования Delphi
  - 1) **Object Pascal**
  - 2) Java
  - 3) C#
4. IDE – это...
  - 1) История развития Delphi
  - 2) Приложение, разрабатываемое программистом

### 3) Интегрированная среда разработки

5. IDE не включает в себя: (несколько вариантов ответа)

- 1) Редактор кода
- 2) Высокопроизводительный компилятор в машинный код
- 3) Объектно-ориентированную модель компонент
- 4) **Эффекты и переходы**
- 5) Визуальное построение приложений
- 6) **Сопровождение ПП**
- 7) Средство для построения баз данных

6. Объектно-ориентированная программа - ...

- 1) Программирование, основанное на объектах
- 2) **Совокупность объектов и способов их взаимодействия**
- 3) Структура среды программирования

7. Установить соответствие

1) Объект	Атрибуты (основные характеристики), которые описывают особенности объекта (цвет, ширина, положение и т.д.)
2) Событие	Совокупность данных (компонентов) и методов работы с ними
3) Свойство	Отклик на внешнее воздействие
<b>- б, 2 – с, 3 - а</b>	

8. Дополнить предложение. Основным окном разрабатываемого приложения является ...

- 1) Код
- 2) **Форма**
- 3) Библиотека
- 4) Объект

9. Дополнить предложение. Коды для стандартных окон диалога и кнопок в системе Delphi получены от ...

- 1) Компилятора
- 2) C++
- 3) **Windows**

10. Названия процедур на определенное событие:

- 1) Компоненты
- 2) **Обработчики событий**
- 3) Редактор кода

11. Назначение главного меню:

- 1) **Осуществление функций управления при разработке программ**
- 2) Сопровождение программных продуктов
- 3) Автоматизирует процесс тестирования программ

12. Назначение панели инструментов:

- 1) Обработка событий
- 2) **Меню команд быстрого доступа к командам, содержащее набор кнопок, функции которых эквивалентны наиболее часто употребляемым командам Главного меню**
- 3) Построение баз данных

13. Назначение палитры компонентов:

- 1) **содержит пиктограммы, которые представляют компоненты VCL, которые можно включить в приложение**
- 2) Подделка подписей
- 3) Управление Paint

14. Дополнить предложение. Главное окно управляет окнами ... (*несколько вариантов ответа*)

- 1) **Инспектор объектов**
- 2) Язык ассемблера
- 3) **Редактора кода**
- 4) **Проектировщика форм**
- 5) Трансляции программы

15. Заготовка разрабатываемого приложения - ...

- 1) Компилятор
- 2) Свойство
- 3) **Окно проектировщика форм**
- 4) Главное окно

16. Действия которые нельзя выполнить с помощью проектировщика форм

- 1) Добавить компоненты в форму
- 2) **Сменить язык программирования**
- 3) Модифицировать форму и её компоненты
- 4) Связать обработчики событий компонента с программой на Object Pascal, содержащейся в редакторе кода
- 5) **Документирование программы**

17. Дополнить предложение. Инспектор объектов позволяет ... (*несколько вариантов ответа*)

- 1) **Устанавливать свойства объектов**
- 2) Изменять структуру программного кода
- 3) Изменять наследование классов объектов
- 4) **Назначать методы обработки событий**

18. Дополнить предложение. Инспектор объектов состоит из ... (*несколько вариантов ответа*)

- 1) Unit – программного модуля



- 2) **Properties** – списка свойств,
  - 3) **Events** – списка событий.
  - 4) Begin...end.
19. Окно содержащее текст программы на языке Object Pascal, связанный с каждой формой в приложении:
- 1) Окно компилятора
  - 2) **Окно редактора кода**
  - 3) Окно проектировщика
  - 4) Главное окно
20. Задание метки текста из редактора ввода:
- 1) Memo1.Caption := Edit1.Text;
  - 2) **Label1. Caption := Edit1.Text;**
  - 3) Form1. Caption := Edit1.Text;
21. Обнуление строки ввода:
- 1) **Edit1.Text :='';**
  - 2) Edit1.Text := TМемо;
  - 3) Edit1.Text :='Закругляемся'
22. Передача фокуса ввода на редактор ввода
- 1) Label1.SetFocus;
  - 2) Edit1.Add;
  - 3) **Edit1.SetFocus;**
23. Открыть файл проекта Project1 можно, нажав:
- 1) CTRL+F4
  - 2) CTRL+ALT+F10
  - 3) **CTRL+F12**
  - 4) ALT+F12
24. Для обозначения комментария не используются:
- 1) (\*комментарий\*)
  - 2) \комментарий/
  - 3) {комментарий}
  - 4) //комментарий
25. Зарезервированное слово, объявляющее блок подключаемых к проекту модулей:
- 1) Program
  - 2) Begin..end
  - 3) **Uses**
  - 4) Forms
26. Дополнить предложение. Объект Application определяет ...
- 1) **Свойства и методы программы-приложения для Windows**
  - 2) Отображение размещения формы

- 3) Перемещение по программному коду
- 27.Метод Application.Run:
- 1) Создает окно формы, регистрирует входящие в него компоненты
  - 2) **Подключает автоматический цикл обработки сообщений Windows к приложению**
  - 3) Выполняет подпрограммы раздела инициализации модулей приложения в случае их наличия
- 28.Контейнер, содержащий все элементы программы:
- 1) Unit1
  - 2) **Form1**
  - 3) Edit1
  - 4) Memo1
- 29.Дополнить предложение. Свойство **Position** определяет ...
- 1) **Размещение и размер формы**
  - 2) Подключение модулей
  - 3) Цвет шрифта
30. **poDesigned:**
- 1) Форма выводится в центр экрана
  - 2) Windows автоматически определяет начальную позицию и размеры формы
  - 3) **Форма отображается в той позиции и с теми размерами, которые были установлены при её конструировании**
- 31.Определяет способ выравнивания компонента внутри контейнера:
- 1) Свойство Position
  - 2) **Свойство Align**
  - 3) Свойство Memo
  - 4) Свойство ScrollBars
32. **alClient:**
- 1) **Компонент занимает всю поверхность контейнера**
  - 2) Выравнивание не используется
  - 3) Компонент помещается в нижнюю часть контейнера
33. **alCustom:**
- 1) Компонент помещается в верхнюю часть контейнера
  - 2) Компонент помещается в нижнюю часть контейнера
  - 3) **Размеры и положение компонента устанавливаются разработчиком**
- 34.Объект **Memo1** - ...
- 1) Однострочный редактор
  - 2) **Многострочный редактор, содержащий несколько строк текста**
  - 3) Текстовая надпись
- 35.Свойство **ScrollBars:**

- 1) **Задаёт в поле редактирования полосы прокрутки**
- 2) Определяет способ выравнивания компонента внутри контейнера
- 3) Задаёт имя объекта

36. Функциональная кнопка с изображением устанавливается из:

- 1) Палитра компонентов -> Standart -> Button
- 2) Палитра компонентов -> Standart -> Label
- 3) **Палитра компонентов -> Additional -> BitBtn**

37. Дополнить предложение. За стиль оформления внешней и внутренней рамок отвечают свойства ... (несколько вариантов ответа)

- 1) Align
- 2) **BevelOuter**
- 3) **BevelInner**
- 4) ssBoth
- 5) Caption

38. Установить соответствие

1) bvLowered	a) Компонент помещается в нижнюю часть контейнера
2) poScreenCenter	b) Есть обе полосы прокрутки
3) bvNone	c) Рамка вдавлена
4) alBottom	d) Форма выводится в центре экрана, её высота и ширина не изменяются
5) ssBoth	e) Рамка отсутствует
<b>-c, 2- d, 3 – e, 4 – a, 5 -b</b>	

39. VCL – это...

- 1) Приложение Delphi
- 2) Алгоритмизация решения задачи
- 3) **Библиотека визуальных компонентов**
- 4) Характеристики объектов

40. Дополнить предложение. Компоненты, которые видны на форме, как во время создания приложения, так и во время работы приложения называются ...

- 1) **Визуальными**
- 2) Реальными
- 3) Невизуальными
- 4) Основными

41. К невидуальным компонентам относят:

- 1) Кнопки, метки, списки блоков
- 2) **Таймеры, компоненты для работы с базами данных, списки изображений**
- 3) Библиотеки, звуки, коды

42. Установить соответствие:

1) Standart	a) Обеспечивает доступ к 32-битным элементам Windows
2) Additional	b) Компоненты, реализующие интерфейс с пользователем и процесс управления данными для БД
3) Win32	c) Включает стандартные компоненты, обеспечивающие некоторые функции интерфейса пользователя
4) System	d) Специализированные компоненты, организующие доступ к БД
5) DataAccess	e) Набор компонентов для доступа к системным ресурсам (OLE, DDE)
6) DataControl	f) Дополнительные интерфейсные компоненты для красочного оформления приложения
<b>1) – c, 2) – f, 3) – a, 4) – e, 5) – d, 6) – b;</b>	

43. Дополнить предложение. Имя компонента состоит из... (несколько вариантов ответа)

- 1) **Названия компонента**
- 2) Заготовки компонента
- 3) **Порядкового номера компонента**
- 4) Произвольного значения ItemIndex

44. Дополнить предложение. Свойства Height и Width показывают...

- 1) Какое действие должен выполнить программист
- 2) **Размеры компонента**
- 3) Исполняемый блок программы

45. Дополнить предложение. Проект Delphi представляет собой...

- 1) **Набор программных единиц – модулей**
- 2) Основной интерфейсный элемент в Delphi
- 3) Атрибуты (основные характеристики), которые описывают особенности объекта

46. Модуль – это...

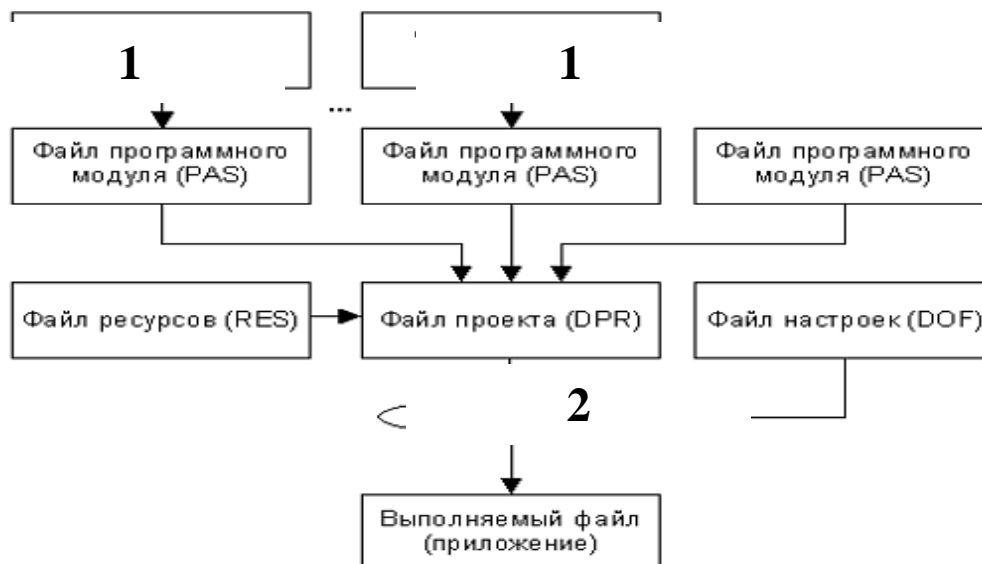
- 1) Средство для создания, изменения исходных файлов, которые содержат написанную на языке программирования программу
- 2) **Автономно компилируемая программная единица, включающая в себя различные компоненты раздела описаний (типы, константы, переменные, процедуры и функции) и, возможно некоторые исполняемые операторы иницилирующей части**
- 3) Набор инструментов, которые используются для преобразования символов

47. Установить соответствие

1) Файл проекта	a) (.pas) соответствующий файл модуль для хранения кода
-----------------	---

2) Файл модуля	b) (.res) содержит пиктограмму и прочие ресурсы
3) Файл формы	c) (.dpr) текстовый файл используется для хранения информации о формах и модулях, содержит операторы инициализации и запуска программы на выполнение
4) Файл опций проекта	d) (.dfm) двоичный файл, который создается для хранения информации о ваших формах и фреймах
5) Файл ресурсов	e) (.cfg) хранит установки проекта
6) Файл конфигурации проекта	f) (.dof) хранит установки опций проекта
<b>1) – c, 2) – a, 3) – d, 4) – f, 5) – b, 6) – e</b>	

48. Заполнить схему:



1) 1 – файл формы (DFM); 2 – файл проекта (DPR)

2) 1 – файл проекта (DPR); 2 – файл формы (DFM)

3) 1 – файл формы (DPR); 2 – файл проекта (DFM)

4) 1 – файл проекта (DFM); 2 – файл формы (DPR)

49. Дополнить предложение. Набор всех файлов, необходимых для создания приложения, называется...

1) Модулем

2) **Проектом**

3) Задачей

4) Формой

50. Файлы описания форм – это...

1) Отклики на внешнее воздействие

- 2) Файлы подключающие все используемые программные модули
  - 3) **Текстовые файлы с расширением DFM, в котором сохраняются значения свойств формы и ее компонентов**
51. Содержимое dfm-файла вызывается:
- 1) **Контекстное меню формы -> View as Text**
  - 2) Контекстное редактора кода -> View as Form
  - 3) File -> Save as...
52. Дополнить предложение. При сборке приложения описание из DFM-файла помещается в...
- 1) Файл модуля
  - 2) **Область ресурсов**
  - 3) Область установок проекта
53. Программный модуль (Unit) содержит:
- 1) **Все относящиеся к форме объявления и методы обработки событий, написанные на языке Object Pascal**
  - 2) Заготовку, макет одного из окон разрабатываемого приложения
  - 3) Пиктограммы, которые представляют компоненты VCL, которые можно включить в приложение
54. Дополнить предложение. Имя модуля должно совпадать с...
- 1) Приложением Windows
  - 2) Спецификацией задачи
  - 3) Объектом в Delphi
  - 4) **Именем файла при сохранении File->Save as...**
55. Интерфейсная секция модуля (interface) содержит: (несколько вариантов ответа)
- 1) **Список стандартных модулей библиотеки VCL**
  - 2) **Раздел описания типов**
  - 3) Компилятор
  - 4) **Раздел описания переменных (объявление собственно объекта формы)**
  - 5) Инспектор объектов и проектировщик форм
56. Дополнить предложение. Класс формы объявляется в разделе...
- 1) **Type**
  - 2) Uses
  - 3) Project
  - 4) Var
57. Дополнить предложение. В описании класса форм помещенные на форму компоненты представлены...
- 1) Границами формы
  - 2) **Полями формы**
  - 3) Индексами формы

58. Published – это...

- 1) **Атрибут видимости, позволяющий работать с компонентами и методами обработки событий на визуальном уровне**
- 2) Директива препроцессора
- 3) Текст программы на языке Object Pascal

59. Пустые секции, в которых можно размещать любые вспомогательные поля, методы и свойства: *(несколько вариантов ответа)*

- 1) **Private**
- 2) Uses
- 3) Var
- 4) **Public**
- 5) Program

60. Объявление объекта формы:

- 1) unit Unit1;
- 2) type TForm1 = class(TForm)
- 3) **var Form1: TForm1;**

61. implementation – это...

- 1) Тип данных Delphi
- 2) Племя в Южной Африке
- 3) **Раздел реализации объявлений интерфейса**

62. Подключение файла описания формы:

- 1) **{R \*.dfm}**
- 2) {R \*.pas}
- 3) {R \*.dpr}

63. Дополнить предложение. После подключения файла описания формы следует...

- 1) Раздел описания типов
- 2) Атрибут видимости
- 3) Описание размеров используемых компонентов
- 4) **Реализация методов обработки событий**

64. Установить метод обработки события Button1Click: *(несколько вариантов ответа)*

- 1) **Двойной щелчок по компоненту Button1 на форме**
- 2) **Выделить Button1 ->Events -> OnClick**
- 3) Выделить BitBtn1 ->Events -> OnClick

65. Программный файл проекта – это...

- 1) Программист
- 2) **Главная часть приложения, с которой начинается выполнение программы и которая обеспечивает инициализацию других модулей**
- 3) Пустая секция

66.К комментариям не относятся: (*несколько вариантов ответа*)

- 1) **Program**
- 2) {Form1 }
- 3) **{R \*.res}**
- 4) //выполнение команды

67.Дополнить предложение. Сочетание end со следующей за ней точкой называется...

- 1) Терминалом программной единицы
- 2) **Терминатором программной единицы**
- 3) Андроидом программной единицы

68.Содержимое DPR-файла вызывается: (*несколько вариантов ответа*)

- 1) Ctrl+alt
- 2) File -> New->Application
- 3) **Project -> View Source**
- 4) **Ctrl+F12**

69.Дополнить предложение. Зарезервированное слово **uses** сообщает компилятору о...

- 1) **Модулях, которые должны рассматриваться как необъемлемые части программы и которые содержатся в других файлах**
- 2) Начале работы приложения
- 3) Ошибках, которые могут возникнуть при компилировании

70.Модуль Forms – это...

- 1) Дополнительный модуль, содержащий неиспользуемые компоненты
- 2) Модуль для построения более красочного приложения
- 3) **Обязательный для всех программ модуль, который содержит определение объекта Application**

71.Запись **{R \*.res}** означает:

- 1) Комментарий, указывающий программисту где искать ресурсы
- 2) **Директиву, подключающую к результирующему выполняемому файлу ресурсы**
- 3) Директиву, подключающую к результирующему выполняемому файлу форму

72.Установить соответствие:

1) Application.Initialize	a) загружает и инициализирует форму
2) Application.CreateForm	b) активизирует форму и начинает выполнение приложения
3) Application.Run	c) подготавливает приложение к работе
<b>– c; 2 – a; 3 – b</b>	

73.Включение множественного выбора элементов компонента ListBox1



- 1) ListBox2.MultiSelect:=false;
- 2) **ListBox1.MultiSelect:=true;**
- 3) ListBox1.MultiSelect:=false;
- 4) ListBox1.Checked:=true;

74. Запись ListBox1.Selected[3]:=true означает что:

- 1) Необходимо установить три дополнительных элемента
- 2) Необходимо удалить три лишних элемента
- 3) **Выделен четвертый элемент в компоненте**
- 4) Выделен третий элемент в компоненте

75. Простейшая функция вывода на экран текстовых сообщений:

- 1) ListBox1.Items.Add();
- 2) **ShowMessage();**
- 3) Case...of;
- 4) CheckBox1.Checked:=true;

76. Полная очистка списка:

- 1) **ListBox1.Clear;**
- 2) ListBox1.Items.Add();
- 3) RadioGroup1.ItemIndex:=0;

77. Снятие «флажка» компонента CheckBox1

- 1) ListBox1.Checked:=true;
- 2) **CheckBox1.Checked:=false;**
- 3) CheckBox1.Checked:=true;
- 4) CheckBox1.Caption := 'Снять флажок';

78. Свойство Align=AllBottom:

- 1) Компонент перемещается в центр формы
- 2) Компонент перемещается в верхнюю часть контейнера
- 3) **Компонент перемещается в нижнюю часть контейнера и его ширина становится равной ширине контейнера**

79. Объект ComboBox - ...

- 1) Многострочный редактор
- 2) **Комбинированный список**
- 3) Функциональная кнопка

80. Установка в ComboBox1 первой строки массива Item:

- 1) **ItemIndex=0**
- 2) ItemIndex=-1
- 3) ItemIndex=1
- 4) ItemIndex=356

81. Объект RadioGroup - ...

- 1) Команда «Авторадио»

2) **Панель группы радиокнопок**

3) Метка

82. Дополнить предложение. Компонент-флажок, который используется для обозначения включения или отключения какой либо опции, называется...

1) `RadioGroup`

2) Функциональная кнопка

3) **`CheckBox`**

4) `ListBox`

Критерии оценки тестовых заданий.

Оценка	Число выполненных заданий
5(отлично)	все
4(хорошо)	выполнено не полностью
3(удовлетворительно)	выполнено частично
2(неудовлетворительно)	не выполнено

### 3.1 Основные источники:

Электронные издания (электронные ресурсы)

1. Войтова, М.В. Информационные технологии в профессиональной деятельности : учеб. пособие / М.В. Войтова . – Москва : ФГБУ ДПО «Учебно-методический центр по образованию на железнодорожном транспорте», 2019. – 128 с. – ISBN 978-5-907055-81-0
2. Трофименко, В. Н. Вычислительная техника и информационные технологии : учебное пособие / В. Н. Трофименко. — Ростов-на-Дону : РГУПС, 2019. — 151 с. — ISBN 978-5-88814-885-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/140609> (дата обращения: 21.10.2020). — Режим доступа: для авториз. пользователей.
3. Назаров, С.В. Операционные системы. Практикум : учебное пособие / Назаров С.В., Гудыно Л.П., Кириченко А.А. — Москва : КноРус, 2020. — 372 с. — (бакалавриат). — ISBN 978-5-406-07707-8. — URL: <https://book.ru/book/933567> (дата обращения: 21.10.2020). — Текст : электронный.
4. Трофименко, В. Н. Вычислительная техника и информационные технологии : учебное пособие / В. Н. Трофименко. — Ростов-на-Дону : РГУПС, 2019. — 151 с. — ISBN 978-5-88814-885-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/140609> (дата обращения: 21.10.2020). — Режим доступа: для авториз. пользователей.

## **1. Контрольно-оценочные материалы для итоговой аттестации**

### **Вопросы для итогового контроля**

1. Основные принципы объектно-ориентированного программирования. Классы: основные понятия.
2. Перегрузка методов.
3. Операции класса.
4. Иерархия классов.
5. Синтаксис интерфейсов.
6. Интерфейсы и наследование.
7. Структуры.
8. Делегаты.
9. Регулярные выражения
10. Коллекции. Параметризованные классы.
11. Указатели
12. Операции со списками
13. Работа с классами.
14. Перегрузка методов.
15. Определение операций в классе.
16. Создание наследованных классов
17. Работа с объектами через интерфейсы.
18. Использование стандартных интерфейсов.
19. Работа с типом данных структура.
20. Коллекции. Параметризованные классы.
21. Использование регулярных выражений
22. Операции со списками.
23. Использование основных шаблонов.
24. Использование порождающих шаблонов.
25. Использование структурных шаблонов.
26. Использование поведенческих шаблонов.
27. Разработка приложения с использованием текстовых компонентов
28. Разработка приложения с несколькими формами.
29. Разработка приложения с не визуальными компонентами.
30. Разработка игрового приложения.
31. Разработка приложения с анимацией.
32. Создание приложения с БД
33. Создание запросов к БД
34. Создание хранимых процедур
35. Тестирование «белым ящиком»
36. Тестирование «черным ящиком»
37. Модульное тестирование
38. Интеграционное тестирование
39. Оформление документации на программные средства с использованием инструментальных средств.
40. Установка инструментария и настройка среды для разработки мобильных

приложений

41. Установка среды разработки мобильных приложений с применением виртуальной машины
42. Создание эмуляторов и подключение устройств»
43. Настройка режима терминала»
44. Создание нового проекта»
45. Изучение и комментирование кода»
46. Лабораторная работа «Изменение элементов дизайна»
47. Обработка событий: подсказки»
48. Обработка событий: цветовая индикация»
49. Подготовка стандартных модулей»
50. Обработка событий: переключение между экранами»
51. Передача данных между модулями»
52. Тестирование и оптимизация мобильного приложения»
53. Использование потоков.
54. Обмен данными.
55. Сетевое программирование сокетов.
56. Работы с буфером экрана.

### **Критерии оценки**

- **оценка «отлично»** выставляется студенту, если дан правильный ответ на 2 теоретических вопроса и выполнены правильно все практические задания;
- **оценка «хорошо»** если дан правильный ответ на 2 теоретических вопроса и выполнено правильно одно практическое задание или дан правильный ответ на теоретический вопрос и выполнены правильно все практические задания;
- **оценка «удовлетворительно»** если дан правильный ответ на теоретический вопрос и выполнено правильно одно практическое задание или дан правильный ответ на 2 теоретических вопроса, или выполнены правильно 2 практических задания;
- **оценка «неудовлетворительно»** если не дан правильный ответ на 2 теоретических вопроса и не выполнены правильно все практические задания.