

Федеральное государственное образовательное бюджетное
учреждение высшего образования
**«Финансовый университет при Правительстве Российской Федерации»
(Финуниверситет)**

**Самарский финансово-экономический колледж
(Самарский филиал Финуниверситета)**

УТВЕРЖДАЮ

Заместитель директора по учебно-
методической работе

Л.А Косенкова

« 21 » февраля 20 22 г.



**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ
«ПМ.01 РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ»**

**СПЕЦИАЛЬНОСТЬ: 09.02.07 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И
ПРОГРАММИРОВАНИЕ**

Самара – 2022

Фонд оценочных средств по профессиональному модулю разработан в соответствии с рабочей программой по профессиональному модулю «Разработка модулей программного обеспечения для компьютерных систем», с федеральным государственным образовательным стандартом среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного приказом Министерства образования и науки Российской Федерации от 09.12.2016 года № 1547, с учетом Профессионального стандарта, утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 11 апреля 2014 г. № 647н «Об утверждении профессионального стандарта 06.011 Администратор баз данных (зарегистрирован Министерством юстиции Российской Федерации 24 ноября 2014 г., регистрационный № 34846

Присваиваемая квалификация: администратор баз данных

Разработчики:

Платковская Е.А.



Преподаватель Самарского филиала Финуниверситета

Чурилов А.С.

Преподаватель Самарского филиала Финуниверситета

Рецензент:

Шарамыгина Т.В.



Директор ООО «Ризотек»

Фонд оценочных средств профессионального модуля рассмотрен и рекомендован к утверждению на заседании предметной (цикловой) комиссии естественно-математических дисциплин

Протокол от « 24 » января 20 22 г. № 5

Председатель ПЦК  М.В. Писцова

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Фонд оценочных средств предназначен для контроля и оценки уровня результатов освоения необходимых знаний и умений обучающихся, осваивающих программу учебной дисциплины ПМ.01 Разработка модулей программного обеспечения для компьютерных систем специальности среднего профессионального образования 09.02.07 Информационные системы и программирование.

В результате освоения профессионального модуля ПМ.01 Разработка модулей программного обеспечения для компьютерных систем.

обучающийся должен освоить соответствующие общие и профессиональные компетенции:

Перечень общих компетенций

Код	Наименование общих компетенций
ОК 01	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
ОК 02	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
ОК 03	Планировать и реализовывать собственное профессиональное и личностное развитие.
ОК 04	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
ОК 05	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.
ОК 06	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей
ОК 07	Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.
ОК 08	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
ОК 09	Использовать информационные технологии в профессиональной деятельности.
ОК 10	Пользоваться профессиональной документацией на государственном и иностранном языке
ОК 11	Планировать предпринимательскую деятельность в профессиональной сфере

Перечень профессиональных компетенций

Код	Наименование видов деятельности и профессиональных компетенций
ВД 1	Разработка модулей программного обеспечения для компьютерных систем
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
ПК 1.5	Осуществлять рефакторинг и оптимизацию программного кода
ПК 1.6	Разрабатывать модули программного обеспечения для мобильных платформ

Перечень применяемых профессиональных стандартов

Наименование профессионального стандарта	Обобщенные трудовые функции	Трудовые функции
Код 06.011 «Администратор баз данных» утвержден приказом Министерства труда и социальной защиты РФ от 17.09.2014 г. №647н (ред. от 12.12.2016 г.)	А. Обеспечение функционирования БД	А/06.4 Мониторинг событий, возникающих в процессе работы БД

Сопоставление единиц ФГОС СПО и профессиональных стандартов

Вид деятельности	Профессиональные компетенции	Общие трудовые функции
		06.011
Разработка модулей программного обеспечения для компьютерных систем	ПК 1.1.	
	ПК 1.2.	
	ПК 1.3.	
	ПК 1.4.	А/06.4
	ПК 1.5	
	ПК 1.6	

В результате оценки осуществляется проверка умений:

- 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

В результате оценки осуществляется проверка знаний:

- 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

**ПАСПОРТ
ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ**

Результаты обучения (освоенные умения, усвоенные знания)	ОК	Наименование темы	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
<p>уметь</p> <ul style="list-style-type: none"> – 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; <p>знать</p> <ul style="list-style-type: none"> – 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 	ОК 09-10 ПК 1.1, ПК 1.2	Раздел 1. Выполнение разработки программных модулей	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена
	ОК 09-10 ПК1.3, ПК 1.4, ПК 1.5	Раздел 2. Поддержка и тестирование программных модулей	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена
	ОК 09-10 ПК 1.2, ПК 1.6	Раздел 3. Разработка мобильных приложений	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена
	ОК 09-10 ПК 1.2, ПК 1.3	Раздел 4. Системное программирование	Вопросы для устного и письменного опроса по теме: Практическое занятие выполнение заданий по теме	Вопросы для проведения экзамена

Материалы для текущего контроля

Типовые задания для оценки освоения МДК 01.01 Разработка программных модулей

Практикум по решению задач

Задача 1. Злоумышленник поменял местами действия в алгоритме вычисления среднего арифметического из квадратного корня трёх чисел:

Присвоить а значение $(a^2+b^2+c^2)/3$.

Вести а,в,с

Сообщить “Среднее арифметическое квадратов равно”

Сообщить а.

Восстановите правильный порядок действий.

Задача 2. Исправьте следующий алгоритм решения уравнения $(x-2)(x+2)=0$:

Присвоить х значение +2.

Сообщить “Корни уравнения равны”.

Сообщить первое значение х.

Сообщить второе значение х.

Задача 3. Автомобиль проехал три участка пути разной длины с разными скоростями. Составьте алгоритм нахождения средней скорости автомобиля.

Задача 4. Запишите в виде алгоритмов правила определения знака:

А) произведения двух действительных чисел;

Б) суммы двух действительных чисел.

6. В записи алгоритма вычисления значения выражения $(x^2-5x+5)/(x^2-4x+3)$

Злоумышленник одно действие поставил не на свое место. Вот как стал выглядеть алгоритм:

вести х

если $x^2-4x+3=0$, то:

сообщить “При таком х значение выражения не определено”.

иначе:

присвоить у значение $(x^2-5x+5)/(x^2-4x+3)$.

конец ветвления.

сообщить у.

Верните действие на свое место.

Задача 5. Вывести на экран 3 строки текста:

Сказка Репка; 2) Посадил дед Репку); 3) Выросла Репка большая-пребольшая.

Задача 6. Вычислить значения корней квадратного уравнения:

$$x^2+px+q=0$$

Тест

1. Алгоритм — это:

а) правила выполнения определенных действий;

б) ориентированный граф, указывающий порядок исполнения некоторого набора команд;

в) понятное и точное предписание исполнителю совершить последовательность действий, направленных на достижение поставленных целей;

- г) набор команд для компьютера;
- д) протокол вычислительной сети.

2. Укажите наиболее полный перечень способов записи алгоритмов:

- а) словесный, графический, псевдокод, программный;
- б) словесный;
- в) графический, программный;
- г) словесный, программный;
- д) псевдокод.

3. Суть такого свойства алгоритма как результативность заключается в том, что:

- а) алгоритм должен иметь дискретную структуру (должен быть разбит на последовательность отдельных шагов);
- б) записывая алгоритм для конкретного исполнителя, можно использовать лишь те команды, что входят в систему его команд;
- в) алгоритм должен обеспечивать решение не одной конкретной задачи, а некоторого класса задач данного типа;
- г) при точном исполнении всех команд алгоритма процесс должен прекратиться за конечное число шагов, приведя к определенному результату;
- д) исполнитель алгоритма не должен принимать решения, не предусмотренные составителем алгоритма.

4. Суть такого свойства алгоритма как массовость заключается в том, что:

- а) алгоритм должен иметь дискретную структуру (должен быть разбит на последовательность отдельных шагов);
- б) записывая алгоритм для конкретного исполнителя, можно использовать лишь те команды, что входят в систему его команд;
- в) алгоритм должен обеспечивать решение не одной конкретной задачи, а некоторого класса задач данного типа;
- г) при точном исполнении всех команд алгоритма процесс должен прекратиться за конечное число шагов, приведя к определенному результату;
- д) исполнитель алгоритма не должен принимать решения, не предусмотренные составителем алгоритма.

5. Суть такого свойства алгоритма как дискретность заключается в том, что:

- а) алгоритм должен иметь дискретную структуру (должен быть разбит на последовательность отдельных шагов);
- б) записывая алгоритм для конкретного исполнителя, можно использовать лишь те команды, что входят в систему его команд;
- в) алгоритм должен обеспечивать решение не одной конкретной задачи, а некоторого класса задач данного типа;
- г) при точном исполнении всех команд алгоритма процесс должен прекратиться за конечное число шагов, приведя к определенному результату;
- д) исполнитель алгоритма не должен принимать решения, не предусмотренные составителем алгоритма.

6. Суть такого свойства алгоритма как понятность заключается в том, что:

- а) алгоритм должен иметь дискретную структуру (должен быть разбит на последовательность отдельных шагов);
- б) записывая алгоритм для конкретного исполнителя, можно использовать лишь те команды, что входят в систему его команд;

- в) алгоритм должен обеспечивать решение не одной конкретной задачи, а некоторого класса задач данного типа;
- г) при точном исполнении всех команд алгоритма процесс должен прекратиться за конечное число шагов, приведя к определенному результату;
- д) исполнитель алгоритма не должен принимать решения, не предусмотренные составителем алгоритма.

7. Суть такого свойства алгоритма как детерминированность заключается в том, что:

- а) алгоритм должен иметь дискретную структуру (должен быть разбит на последовательность отдельных шагов);
- б) записывая алгоритм для конкретного исполнителя можно использовать лишь те команды, что входят в систему его команд;
- в) алгоритм должен обеспечивать решение не одной конкретной задачи, а некоторого класса задач данного типа;
- г) при точном исполнении всех команд алгоритма процесс должен прекратиться за конечное число шагов, приведя к определенному результату;
- д) исполнитель алгоритма не должен принимать решения, не предусмотренные составителем алгоритма.

8. Алгоритм называется линейным:

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

9. Алгоритм называется циклическим:

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

10. Алгоритм включает в себя ветвление, если:

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

11. Системы программирования:

- а) обеспечивают непосредственное решение пользовательских задач;
- б) позволяют создавать новые программы на языках программирования;
- в) обеспечивают работу всех аппаратных устройств компьютера и доступ пользователя к ним;

- г) представляют собой совокупность программ, используемых для различных операций с документами;
- д) предназначены для уничтожения компьютерных вирусов и зараженных ими файлов.

12. Характерным признаком линейной программы является:

- а) строго последовательное выполнение операторов в порядке их записи;
- б) наличие в каждой программной строке только одного операторов;
- в) использование в ней исключительно операторов присваивания;
- г) присутствие в ней операторов условного и безусловного перехода;
- д) присутствие в ней операторов цикла.

13. Подпрограммой называют:

- а) часть программы, содержащую неоднократно выполняемые команды;
- б) независимый программный модуль;
- в) произвольный фрагмент программы;
- г) набор операторов, следующий в программе за оператором GOSUB;
- д) часть программы, служащей для решения некоторой вспомогательной задачи.

14. Важнейший принцип структурного программирования базируется на утверждении:

- а) любой алгоритм имеет дискретную структуру;
- б) алгоритм любой сложности можно построить с помощью следующих базовых структур: линейной, ветвящейся, циклической;
- в) современный компьютер — это единство аппаратных средств и программного обеспечения;
- г) сущность формализации решаемой задачи заключается в составлении алгоритма;
- д) в качестве обязательного этапа создания программы выступает ее тестирование и отладка.

15. Переменная в программировании полностью характеризуется:

- а) именем;
- б) именем, значением и типом;
- в) именем и типом;
- г) именем и значением;
- д) значением.

Ключ к тесту:

- | | | | | | |
|----|-------|-----|---------|-----|-------|
| 1. | а | 7. | а | 13. | а |
| 2. | а | 8. | в | 14. | а б |
| 3. | а б в | 9. | а б в | 15. | а б в |
| 4. | а б в | 10. | б | | |
| 5. | а | 11. | 12345 | | |
| 6. | а | 12. | а б в д | | |

Критерии оценивания работ:

90-100% выполненной работы (34-38 заданий) – «5»

75-85% выполненной работы (28-33 заданий) – «4»

50-70% выполненной работы (19-27 заданий) – «3»

ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ

Промежуточная аттестация по ПМ. 01

Задачи для промежуточной аттестации по ПМ.01

Разработать приложение, позволяющее производить конвертацию валюты. Список валют и их курс хранятся в текстовом файле. Данные из текстового файла считываются в массив, состоящий из записей, содержащих название валюты и курс. Пользователь выбирает валюту из списка → название валюты и курс появляются в поля Label. В поле Edit пользователь вводит сумму в рублях. Далее после щелчка по кнопке «Рассчитать» программа выводит результат

1. Разработать приложение «Тест », состоящий из 10 вопросов и выставляющий оценку. Вопросы и варианты ответов, считываются из текстового файла
2. Разработать приложение, позволяющее придумывать сложный пароль по фамилии, имени, отчеству и году рождения. Алгоритм кодирования придумать самостоятельно. Данные для формирования паролей считываются с текстового файла. Пароли сохраняются в другой текстовый файл.
3. Разработать программу «Пароль». При загрузке программы запрашивает логин и пароль. Если пользователь не зарегистрирован, предлагает зарегистрироваться. Организовать хранение логинов и паролей пользователей в текстовом файле, который загружает данные в массив при запуске приложения.
4. Разработать программу, которая формирует ведомость успеваемости студента. Фамилия и имя студента считываются из текстового файла. Для каждого студента вводятся 3 оценки, рассчитывается средний балл и вся информация (Фамилия, имя, оценка1, оценка2, оценка3, средний_балл) сохраняется в другой текстовый файл по выбору пользователя.
5. Разработать приложение, которое переводит числа из десятичной системы счисления в 2сс, 8сс, 16 сс
6. Разработать приложение, которое переводит числа из 2сс, 8сс, 16 сс в десятичную
7. Разработать приложение «Двоичный калькулятор», позволяющее производить операции сложения, вычитания двоичных чисел
8. Разработать приложение «Восьмеричный калькулятор», позволяющее производить операции сложения, вычитания восьмеричных чисел
9. Разработать приложение «Шестнадцатеричный калькулятор», позволяющее производить операции сложения, вычитания шестнадцатеричных чисел
10. Разработать приложение, позволяющее придумывать сложный пароль по фамилии, имени, отчеству и году рождения. Алгоритм кодирования придумать самостоятельно
11. Разработать приложение «Тест », состоящий из 10 вопросов и выставляющий оценку
12. Разработать приложение «Бегающий текст», которое передаёт текстовое сообщение, введённое в поле EDIT между компонентами формы по нажатию на клавишу Enter, пока не вернётся в поле EDIT.

13. Разработать приложение «Поймай кнопку...». На форме разместить кнопку (можно кнопку заменить рисунком) и при подведении указателя мыши к кнопке, она «убегала» или «пряталась»
14. Составить проект для нахождения корней квадратного уравнения по трем его коэффициентам, вводимым с клавиатуры. Предусмотреть реакции на некорректный ввод символов в полях редактирования (например, поставить защиту от ввода букв или второй десятичной точки). При нажатии Enter переносить курсор в следующее поле редактирования (а на завершении ввода выполнять вычисления). Предусмотреть кнопки "расчет", "новые данные" и "выход". Всем введенным компонентам задать ярлычки с оперативной подсказкой (Hints). При оформлении компонент использовать по возможности различные цвета и шрифты.
15. Составить проект для визуализации выбираемого стиля, размера и цвета шрифта. Сам шрифт как набор всех латинских и русских букв (как прописных, так и строчных) отображать на метке. Каждую характеристику шрифта выбирать из набора минимум четырех радиокнопок.
16. Составить проект для анализа введенной в строке Edit информации: текстовая, числовая, прочая. В качестве индикаторов использовать набор из трех компонент CheckBox. Ввести четвертый индикатор для анализа очередного набираемого символа.
17. Предусмотреть кнопки "новые данные" и "выход". Всем введенным компонентам задать ярлычки с оперативной подсказкой (Hints). При оформлении компонент использовать по возможности различные цвета и шрифты.

Промежуточная аттестация МДК 01.01

Тест 1.

1. Операционная система:
 - а) система программ, которая обеспечивает совместную работу всех устройств компьютера по обработке информации**
 - б) система математических операций для решения отдельных задач
 - в) система планового ремонта и технического обслуживания компьютерной техники
2. Программное обеспечение (ПО) – это:
 - а) совокупность программ, позволяющих организовать решение задач на компьютере**
 - б) возможность обновления программ за счет бюджетных средств
 - в) список имеющихся в кабинете программ, заверен администрацией школы
3. Загрузка операционной системы – это:
 - а) запуск специальной программы, содержащей математические операции над числами
 - б) загрузка комплекса программ, которые управляют работой компьютера и организуют диалог пользователя с компьютером**
 - в) вложение дискеты в дисковод
4. Система программирования – это:
 - а) комплекс любимых программ программиста
 - б) комплекс программ, облегчающий работу программиста**
 - в) комплекс программ, обучающих начальным шагам программиста
5. Прикладное программное обеспечение – это:
 - а) 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, PowerPoint
 - 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) Задает начальное значение полям
 - d) Выделяет память, задает начальное значение полям, выполняет любые проверки, заданные программистом
16. Как описывается конструктор объекта
- a) procedurecreate;
 - b) constructorcreate;
 - c) functioncreate;
 - d) functionconstructor;**
17. Как описывается деструктор объекта
- a) procedurefree;
 - b) destructorfree;**
 - c) free;
 - d) functionfree;
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 «черный ящик» являются
- a) Модели мышления
 - b) Модели, описывающие зависимость параметров состояния объекта от входных параметров
 - c) Модели, описывающие входные и выходные параметры объекта без учета 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 учетом 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
- Процесс, при котором от начального предположения осуществляется движение по направлению к лучшим решениям
 - Процесс пошагового разбиения алгоритма на все более мелкие части с целью получения таких элементов, для которых можно написать конкретные команды**
 - Метод сведения трудной задачи к последовательности более простых
 - Исследование древовидной модели пространства решений и ориентация на поиск оптимального решения
26. Загрузочный модуль программы – результат работы
- Грамматики
 - Транслятора**
 - Интерпретатора
 - Редактора связей (компоновщика)
27. Интегрированная система программирования включает компонент для перевода исходного текста программы в машинный код, который называется
- построителем кода
 - компилятор**
 - переводчиком
 - преобразователем
28. Результатом компиляции программы на языке высокого уровня является
- Командный файл
 - Объектный файл**
 - Исходный текст программы на языке высокого уровня
 - Дисплейный файл

Ключ к тесту

1. a	11. c	21. c
2. a	12. b	22. b
3. b	13. d	23. b c
4. b	14. a	24. d
5. b	15. a	25. b
6. b	16. d	26. b
7. a	17. b	27. b
8. a	18. b	28. b
9. c	19. c	
10. b	20. b	

Критерии оценивания работ:

90-100% выполненной работы (34-38 заданий) – «5»

75-85% выполненной работы (28-33 заданий) – «4»

50-70% выполненной работы (19-27 заданий) – «3»

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

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 := TMemo;
 - 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. Резервированное 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
1 – c, 2- d, 3 – e, 4 – a, 5 -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 невидимым 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
1) – c, 2) – f, 3) – a, 4) – e, 5) – d, 6) – b;	

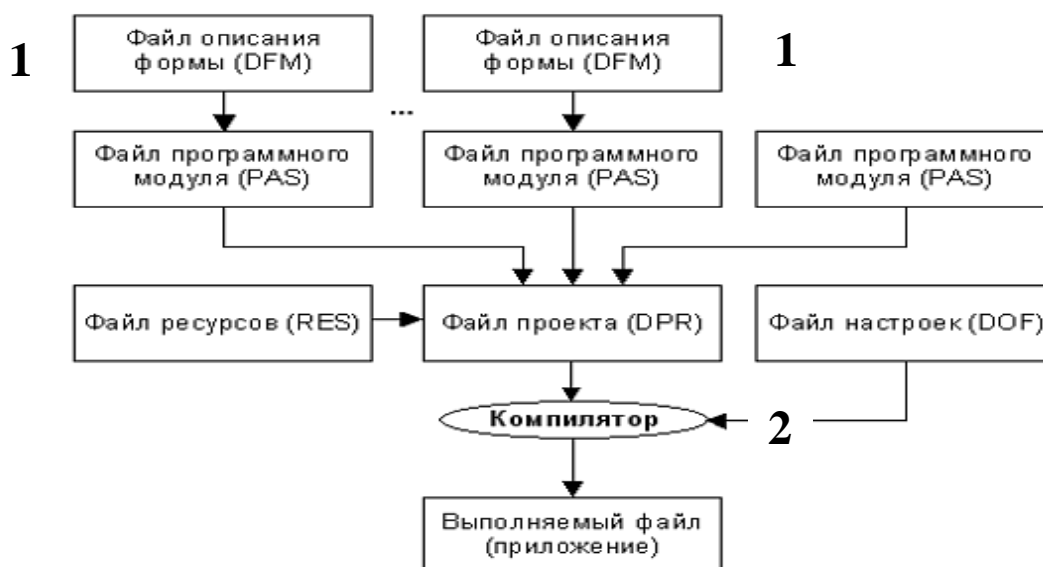
43. Дополнить предложение. Имя компонента состоит из... (несколько вариантов ответа)

- 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
1) – c, 2) – a, 3) – d, 4) – f, 5) – b, 6) – e	

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 ->ViewasText**
 - 2) Контекстное редактора кода ->ViewasForm
 - 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->Saveas...**
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
1 – c; 2 – a; 3 – 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`

Ключ к тесту

1. 1	10. 3	19. 3
2. 1	11. 4	20. 3
3. 3	12. 2	21. 2
4. 4	13. 2	22. 4
5. 3	14. 1	23.
6. 2	15. 3	24.
7. 2	16. 2	25.
8. 1	17. 1	26.
9. 2	18. 4	27.

28.	4) – f, 5)	67.
29.	– b, 6) - e	68.
30.	48.	69.
31.	49.	70.
32.	50.	71.
33.	51.	72. 1-a, 2-c,3-
34.	52.	b,4-d
35.	53.	73.
36.	54.	74.
37.	55.	75.
38.	56.	76.
39.	57.	77.
40.	58.	78.
41.	59.	79.
42.	60.	80.
43.	61.	81.
44.	62.	82.
45.	63.	
46.	64.	
47. – c, 2) –	65.	
a, 3) – d,	66.	

Критерии оценивания работ:

90-100% выполненной работы (34-38 заданий) – «5»

75-85% выполненной работы (28-33 заданий) – «4»

50-70% выполненной работы (19-27 заданий) – «3»

Промежуточная аттестация по МДК 01.02

Тест

1. Какие программы можно отнести к системному программному обеспечению:

Варианты ответа:

- 1) **операционные системы;**
- 2) прикладные программы;
- 3) игровые программы.

2. Какие программы можно отнести к системному ПО:

Варианты ответа:

- 1) **драйверы;**
- 2) текстовые редакторы;
- 3) электронные таблицы;
- 4) графические редакторы.

3. Специфические особенности ПО как продукта:

- 1) *продажа по ценам ниже себестоимости (лицензирование);*
- 2) **низкие материальные затраты при создании программ;**
- 3) **возможность создание программ небольшие коллективом или даже одним человеком;**
- 4) **разнообразие решаемых задач с помощью программных средств.**

4. Какие программы можно отнести к системному ПО:

Варианты ответа:

- 1) программа расчета заработной платы;
- 2) электронные таблицы;
- 3) **СУБД (системы управления базами данных).**

5. Какие программы нельзя отнести к системному ПО:

Варианты ответа:

- 1) **игровые программы;**
- 2) компиляторы языков программирования;
- 3) операционные системы;
- 4) системы управления базами данных.

6. Какие программы можно отнести к прикладному программному обеспечению:

Варианты ответа:

- 1) **электронные таблицы;**
- 2) таблицы решений;
- 3) **СУБД (системы управления базами данных).**

7. Какие программы можно отнести к прикладному ПО:

Варианты ответа:

- 1) **программа расчета заработной платы;**
- 2) диспетчер программ;
- 3) программа «Проводник» (Explorer).

8. Какие программы нельзя отнести к прикладному ПО:

Варианты ответа:

- 1) **компиляторы и (или) интерпретаторы;**
- 2) текстовые и (или) графические редакторы;
- 3) электронные таблицы.

9. Можно ли отнести операционную систему к программному обеспечению:

Варианты ответа:

- 1) **да;**
- 2) нет.

10. Можно ли отнести операционную систему к прикладному программному обеспечению:

Варианты ответа:

- 1) **да;**
- 2) *нет.*

11. Специфические особенности ПО как продукта:

Варианты ответа:

- 1) **низкие затраты при дублировании;**
- 2) универсальность;
- 3) простота эксплуатации;
- 4) наличие поддержки (сопровождения) со стороны разработчика.

12. Какие программы можно отнести к системному ПО:

Варианты ответа:

- 1) **утилиты;**
- 2) экономические программы;
- 3) статистические программы;
- 4) мультимедийные программы.

13. Этап, занимающий наибольшее время, в жизненном цикле программы:

Варианты ответа:

- 1) **сопровождение;**
- 2) проектирование;
- 3) тестирование;
- 4) программирование;
- 5) формулировка требований.

14. Этап, занимающий наибольшее время, при разработке программы:

Варианты ответа:

- 1) **тестирование;**
- 2) сопровождение;
- 3) проектирование;
- 4) программирование;
- 5) формулировка требований.

15. Первый этап в жизненном цикле программы:

Варианты ответа:

- 1) **формулирование требований;**
- 2) анализ требований;
- 3) проектирование;
- 4) автономное тестирование;
- 5) комплексное тестирование.

16. Один из необязательных этапов жизненного цикла программы:

Варианты ответа:

- 1) **оптимизация;**
- 2) проектирование;
- 3) тестирование;
- 4) программирование;
- 5) анализ требований.

17. Самый большой этап в жизненном цикле программы:

Варианты ответа:

- 1) **эксплуатация;**
- 2) изучение предметной области;

- 3) программирование;
- 4) тестирование;
- 5) корректировка ошибок.

18. Какой этап выполняется раньше:

Варианты ответа:

- 1) **отладка;**
- 2) тестирование.

19. Какой этап выполняется раньше:

Варианты ответа:

- 1) отладка;
- 2) оптимизация;
- 3) **программирование;**
- 4) тестирование.

20. Что выполняется раньше:

Варианты ответа:

- 1) **компиляция;**
- 2) отладка;
- 3) компоновка;
- 4) тестирование.

21. Что выполняется раньше:

Варианты ответа:

- 1) **проектирование;**
- 2) программирование;
- 3) отладка;
- 4) тестирование.

22. В стадии разработки программы не входит:

Варианты ответа:

- 1) **автоматизация программирования;**
- 2) постановка задачи;
- 3) составление спецификаций;
- 4) эскизный проект;
- 5) тестирование.

23. Самый важный критерий качества программы:

Варианты ответа:

- 1) **работоспособность;**
- 2) надежность;
- 3) эффективность;
- 4) быстродействие;
- 5) простота эксплуатации.

24. Способы оценки качества:

Варианты ответа:

- 1) **сравнение с аналогами;**
- 2) наличие документации;
- 3) оптимизация программы;
- 4) структурирование алгоритма.

25. Существует ли связь между эффективностью и оптимизацией программы:

Варианты ответа:

- 1) **да;**
- 2) нет.

26. Наиболее важный критерий качества:

Варианты ответа:

- 1) **надежность;**
- 2) быстродействие;
- 3) удобство в эксплуатации;
- 4) удобный интерфейс;
- 5) эффективность.

27. Способы оценки надежности:

Варианты ответа:

- 1) **тестирование;**
- 2) сравнение с аналогами;
- 3) трассировка;
- 4) оптимизация.

28. Повышает ли качество программ оптимизация:

Варианты ответа:

- 1) **да;**
- 2) нет.

29. Существует ли связь между надежностью и быстродействием:

Варианты ответа:

- 1) **нет;**
- 2) да.

30. В каких единицах можно измерить надежность:

Варианты ответа:

- 1) **отказов/час;**
- 2) км/час;
- 3) Кбайт/сек;
- 4) операций/сек.

31. В каких единицах можно измерить быстродействие:

Варианты ответа:

- 1) отказов/час;
- 2) км/час;
- 3) Кбайт/сек;
- 4) **операций/сек.**

32. Что относится к этапу программирования:

Варианты ответа:

- 1) **написание кода программы;**
- 2) В) разработка интерфейса;
- 3) С) работоспособность;
- 4) анализ требований.

33. Последовательность этапов программирования:

Варианты ответа:

- 1) **компилирование, компоновка, отладка;**
- 2) В) компоновка, отладка, компилирование;
- 3) отладка, компилирование, компоновка;
- 4) компилирование, отладка, компоновка.

34) Инструментальные средства программирования:

Варианты ответа:

- 1) **компиляторы, интерпретаторы;**
- 2) СУБД (системы управления базами данных);
- 3) BIOS (базовая система ввода-вывода);
- 4) ОС (операционные системы).

35. На языке программирования составляется:

Варианты ответа:

- 1) **исходный код;**
- 2) исполняемый код;
- 3) объектный код;
- 4) алгоритм.

36. Правила, которым должна следовать программа это:

Варианты ответа:

- 1) **алгоритм;**
- 2) структура;
- 3) спецификация;
- 4) состав информации.

37. Можно ли внутри цикла поместить еще один цикл:

Варианты ответа:

- 1) **да;**
- 2) нет.

38. Можно ли внутри условного оператора поместить еще одно условие:

Варианты ответа:

- 1) **да;**
- 2) нет.

39. Можно ли одно большое (длинное) выражение разбить на несколько выражений:

Варианты ответа:

- 1) **да;**
- 2) нет.

40. Если имеется стандартная функция, нужно ли писать собственную:

Варианты ответа:

- 1) **нет;**
- 2) да.

41. Доступ, при котором записи файла читаются в физической последовательности, называется:

Варианты ответа:

- 1) **последовательным;**
- 2) прямым;
- 3) простым;
- 4) основным.

42. Доступ, при котором записи файла обрабатываются в произвольной последовательности, называется:

Варианты ответа:

- 1) **прямым;**
- 2) последовательным;
- 3) простым;
- 4) основным.

43. Методы программирования (укажите НЕ верный ответ):

Варианты ответа:

- 1) **логическое;**
- 2) структурное;
- 3) модульное.

44. Что выполняется раньше:

Варианты ответа:

- 1) **разработка алгоритма;**
- 2) выбор языка программирования;

- 3) написание исходного кода;
- 4) компиляция.

45. Можно ли переменным присваивать произвольные идентификаторы:

Варианты ответа:

- 1) **да;**
- 2) нет.

46. Найдите НЕ правильное условие для создания имен:

Варианты ответа:

- 1) **имена могут содержать пробелы;**
- 2) длинное имя можно сократить;
- 3) из имени лучше выбрасывать гласные;
- 4) можно использовать большие буквы.

47. Какие символы не допускаются в именах переменных:

Варианты ответа:

- 1) **пробелы;**
- 2) цифры;
- 3) подчеркивание

48. Можно ли использовать имена, которые уже были использованы в другой программе (модуле):

Варианты ответа:

- 1) **да;**
- 2) нет.

49. Можно ли ставить знак подчеркивания в начале имени:

Варианты ответа:

- 1) **да, но не рекомендуется;**
- 2) да, без ограничений;
- 3) нет.

50. Как называется способ составления имен переменных, когда в начале имени сообщается тип переменной:

Варианты ответа:

- 1) прямым указанием;
- 2) **венгерской нотацией;**
- 3) структурным программированием;
- 4) поляризацией.

51. Можно ли писать комментарии в отдельной строке:

Варианты ответа:

- 1) **да;**
- 2) нет.

52. Транслируются ли комментарии:

Варианты ответа:

- 1) да;
- 2) **нет.**

53. Наличие комментариев позволяет:

Варианты ответа:

- 1) **быстрее найти ошибки в программе;**
- 2) быстрее писать программы;
- 3) быстрее выполнять программы.

54. Наличие комментариев позволяет:

Варианты ответа:

- 1) **легче разобраться в программе;**
- 2) применять сложные структуры;
- 3) увеличить быстродействие.

55. Наличие комментариев позволяет:

Варианты ответа:

- 1) **улучшить читабельность программы;**
- 2) улучшить эксплуатацию программы;
- 3) повысить надежность программы.

56. Что определяет выбор языка программирования:

Варианты ответа:

- 1) **область приложения;**
- 2) знание языка;
- 3) наличие дополнительных библиотек.

57. Возможно ли комбинирование языков программирования в рамках одной задачи:

Варианты ответа:

- 1) **да;**
- 2) нет.

58. Допустимо ли комбинирование языков программирования в рамках одной задачи :

Варианты ответа:

- 1) **да;**
- 2) нет.

59. Для каких задач характерно использование большого количества исходных данных, выполнение операций поиска, группировки:

Варианты ответа:

- 1) **для экономических задач;**
- 2) для системных задач;
- 3) для инженерных задач.

60. Для каких задач характерен большой объем вычислений, использование сложного математического аппарата:

Варианты ответа:

- 1) **для инженерных задач;**
- 2) для системных задач;
- 3) для экономических задач.

61. На каком этапе производится выбор языка программирования:

Варианты ответа:

- 1) **проектирование;**
- 2) программирование;
- 3) отладка;
- 4) тестирование.

62. Можно ли использовать комбинацию языков программирования в рамках одного проекта:

Варианты ответа:

- 1) **да;**
- 2) нет.

63. Для решения экономических задач характерно применение:

Варианты ответа:

- 1) **СУБД (систем управления базами данных);**

- 2) языков высокого уровня;
- 3) языков низкого уровня;
- 4) применение сложных математических расчетов.

64. Для решения инженерных задач характерно применение:

Варианты ответа:

- 1) **САПР (систем автоматизированного проектирования);**
- 2) СУБД (систем управления базами данных);
- 3) ОС (операционных систем).

65. Причины синтаксических ошибок:

Варианты ответа:

- 1) **плохое знание языка программирования;**
- 2) ошибки в исходных данных;
- 3) ошибки, допущенные на более ранних этапах;
- 4) неправильное применение процедуры тестирования.

66. Когда можно обнаружить синтаксические ошибки:

Варианты ответа:

- 1) **при компиляции;**
- 2) при отладке;
- 3) при тестировании;
- 4) на этапе проектирования;
- 5) при эксплуатации.

67. Ошибки компоновки заключаются в том, что:

Варианты ответа:

- 1) **указано внешнее имя, но не объявлено;**
- 2) неправильно использовано зарезервированное слово;
- 3) составлено неверное выражение;
- 4) указан неверный тип переменной.

68. Могут ли проявиться ошибки при изменении условий эксплуатации:

Варианты ответа:

- 1) **да;**
- 2) нет.

69. Могут ли проявиться ошибки при изменении в предметной области:

Варианты ответа:

- 1) **да;**
- 2) нет.

70. Возможно ли программирование с защитой от ошибок:

Варианты ответа:

- 1) **да;**
- 2) нет.

71. Есть ли недостатки программирования с защитой от ошибок:

Варианты ответа:

- 1) **да;**
- 2) нет.

72. Защитное программирование это:

Варианты ответа:

- 1) **встраивание в программу отладочных средств;**
- 2) создание задач защищенных от копирования;
- 3) разделение доступа в программе;
- 4) использование паролей;
- 5) оформление авторских прав на программу.

73. Вид ошибки с неправильным написанием служебных слов (операторов):

Варианты ответа:

- 1) **синтаксическая;**
- 2) семантическая;
- 3) логическая;
- 4) символьная.

74. Вид ошибки с неправильным использованием служебных слов (операторов):

Варианты ответа:

- 1) **семантическая;**
- 2) синтаксическая;
- 3) логическая;
- 4) символьная.

75. Ошибки при написании программы бывают:

Варианты ответа:

- 1) **синтаксические;**
- 2) орфографические;
- 3) лексические;
- 4) фонетические;
- 5) морфологические.

76. Процедура поиска ошибки, когда известно, что она есть это:

Варианты ответа:

- 1) **отладка;**
- 2) тестирование;
- 3) компоновка;
- 4) транзакция;
- 5) трансляция.

77. Программа для просмотра значений переменных при выполнении программы:

Варианты ответа:

- 1) **отладчик;**
- 2) компилятор;
- 3) интерпретатор;
- 4) трассировка;
- 5) тестирование.

78. Отладка – это:

Варианты ответа:

- 1) **процедура поиска ошибок, когда известно, что ошибка есть;**
- 2) определение списка параметров;
- 3) правило вызова процедур (функций);
- 4) составление блок-схемы алгоритма.

79. Когда программист может проследить последовательность выполнения команд программы:

Варианты ответа:

- 1) **при трассировке;**
- 2) при тестировании;
- 3) при компиляции;
- 4) при выполнении программы;
- 5) при компоновке.

80. На каком этапе создания программы могут появиться синтаксические ошибки:

Варианты ответа:

- 1) **программирование;**
- 2) проектирование;
- 3) анализ требований;
- 4) тестирование.

81. Когда приступают к тестированию программы:

Варианты ответа:

- 1) **когда программа уже закончена;**
- 2) после постановки задачи;
- 3) на этапе программирования;
- 4) на этапе проектирования;
- 5) после составления спецификаций,

82. Тестирование бывает:

Варианты ответа:

- 1) **автономное;**
- 2) инструментальное;
- 3) визуальное;
- 4) алгоритмическое.

83. Тестирование бывает:

Варианты ответа:

- 1) **комплексное;**
- 2) инструментальное;
- 3) визуальное;
- 4) алгоритмическое.

84. Существует ли различие между отладкой и тестированием:

Варианты ответа:

- 1) **да;**
- 2) нет.

85. При комплексном тестировании проверяются:

Варианты ответа:

- 1) **согласованность работы отдельных частей программы;**
- 2) правильность работы отдельных частей программы;
- 3) быстроедействие программы;
- 4) эффективность программы.

86. Чему нужно уделять больше времени, чтобы получить хорошую программу:

Варианты ответа:

- 1) **тестированию;**
- 2) программированию;
- 3) отладке;
- 4) проектированию.

87. Процесс исполнения программы с целью обнаружения ошибок:

Варианты ответа:

- 1) **тестирование;**
- 2) кодирование;
- 3) сопровождение;
- 4) проектирование.

88. Автономное тестирование это:

Варианты ответа:

- 1) **тестирование отдельных частей программы;**
- 2) инструментальное средство отладки;
- 3) составление блок-схем;
- 4) пошаговая проверка выполнения программы.

89. Трассировка это:

Варианты ответа:

- 1) **проверка пошагового выполнения программы;**
- 2) тестирование исходного кода;
- 3) отладка модуля;

4) составление блок-схемы алгоритма.

90. Локализация ошибки:

Варианты ответа:

- 1) **определение места возникновения ошибки;**
- 2) определение причин ошибки;
- 3) обнаружение причин ошибки;
- 4) исправление ошибки.

91. Назначение тестирования:

Варианты ответа:

- 1) **повышение надежности программы;**
- 2) обнаружение ошибок;
- 3) повышение эффективности программы;
- 4) улучшение эксплуатационных характеристик;
- 5) приведение программы к структурированному виду.

92. Назначение отладки:

Варианты ответа:

- 1) **поиск причин существующих ошибок;**
- 2) поиск возможных ошибок;
- 3) составление спецификаций;
- 4) разработка алгоритма.

93. Инструментальные средства отладки (НЕ правильный ответ):

Варианты ответа:

- 1) **компиляторы;**
- 2) отладчики;
- 3) трассировка.

94. Отладка программ это:

Варианты ответа:

- 1) **локализация и исправление ошибок;**
- 2) алгоритмизация программирования;
- 3) компиляция и компоновка.

95. Что выполняется раньше, автономная или комплексная отладка:

Варианты ответа:

- 1) **автономная;**
- 2) комплексная.

96. Что выполняется раньше, отладка или тестирование:

Варианты ответа:

- 1) **отладка;**
- 2) тестирование.

97. Что такое автоматизация программирования:

Варианты ответа:

- 1) **создание исходного кода программными средствами;**
- 2) создание исходного кода при помощи компилятора;
- 3) создание исходного кода без разработки алгоритма.

98. В чем сущность автоматизации программирования:

Варианты ответа:

- 1) **создание программы без написания ее текста;**
- 2) получение готовой программы без выполнения компоновки;
- 3) в отсутствии компиляции.

99. Возможна ли автоматизация программирования:

Варианты ответа:

- 1) **да;**
- 2) нет.

100. Создание исполняемого кода программы без написания исходного кода называется:

Варианты ответа:

- 1) составлением спецификаций;
- 2) отладкой;
- 3) проектированием.
- 4) **автоматизацией программирования;**

101. Одно из преимуществ автоматизации программирования:

Варианты ответа:

- 1) **наглядное программирование с визуальным контролем;**
- 2) получение стандартной программы;
- 3) создание программы с оптимальным кодом.

102. Один из методов автоматизации программирования:

Варианты ответа:

- 1) структурное программирование;
- 2) модульное программирование;
- 3) **визуальное программирование;**
- 4) объектно-ориентированное программирование.

103. Влияет ли автоматизация программирования на эффективность программы:

Варианты ответа:

- 1) **нет;**
- 2) да

104. Автоматизация программирования позволяет:

Варианты ответа:

- 1) повысить надежность программы;
- 2) **сократить время разработки программы;**
- 3) повысить быстродействие программы.

105. Позволяет ли автоматизация программирования всегда создавать эффективные программы:

Варианты ответа:

- 1) да.
- 2) **нет;**

106. Позволяет ли автоматизация программирования всегда создавать надежные программы:

Варианты ответа:

- a) **нет;**
- b) да.

107. Недостаток автоматизации программирования;

Варианты ответа:

- a) низкое быстродействие;
- b) **большой размер программы;**
- c) сложность программы.

108. Возможны ли ошибки при автоматизации программирования:

Варианты ответа:

- a) **да;**

b) нет.

109. Нужно ли выполнять тестирование при автоматизации программирования:

Варианты ответа:

a) да;

b) нет.

110. Выполняется ли процедура компиляции при автоматизации программирования:

Варианты ответа:

A) да;

B) нет.

111. Что легко поддается автоматизации:

Варианты ответа:

A) интерфейс;

B) работа с файлами;

C) сложные логические задачи;

D) алгоритмизация.

112. Относится ли визуальное программирование к средствам автоматизации:

Варианты ответа:

A) да;

B) нет.

113. Нахождение наилучшего варианта из множества возможных:

Варианты ответа:

A) оптимизация;

B) тестирование;

C) автоматизация;

D) отладка;

E) сопровождение.

114. Что такое оптимизация программ:

Варианты ответа:

A) улучшение работы существующей программы;

B) создание удобного интерфейса пользователя;

C) разработка модульной конструкции программы;

D) применение методов объектно-ориентированного программирования.

115. Критерии оптимизации:

Варианты ответа:

A) время выполнения или размер требуемой памяти;

B) размер программы и ее эффективность;

C) независимость модулей;

D) качество программы, ее надежность.

116. Критерии оптимизации:

Варианты ответа:

A) эффективность использования ресурсов;

B) структурирование алгоритма;

C) структурирование программы.

117. Возможна ли оптимизация программ без участия программиста:

Варианты ответа:

A) да;

B) нет.

118. Возможна ли оптимизация циклов:

Варианты ответа:

- A) да;
- B) нет.

119. В чем заключается оптимизация условных выражений:

Варианты ответа:

- A) **в изменении порядка следования элементов выражения;**
- B) в использовании простых логических выражений;
- C) в использовании сложных логических выражений;
- D) в использовании операций AND, OR и NOT.

120. Оптимизация циклов заключается в:

Варианты ответа:

- A) **уменьшении количества повторений тела цикла;**
- B) просмотре задачи с другой стороны;
- C) упрощение задачи за счет включения логических операций.

121. Оптимизация программы это:

Варианты ответа:

- A) **модификация;**
- B) отладка;
- C) повышение сложности программы;
- D) уменьшение сложности программы.

122. Критерии оптимизации программы:

Варианты ответа:

- A) **быстродействие или размер программы;**
- B) быстродействие и размер программы;
- C) надежность или эффективность;
- D) надежность и эффективность.

123. Результат оптимизации программы:

Варианты ответа:

- A) **эффективность;**
- B) надежность;
- C) машино-независимость;
- D) мобильность.

124. Сущность оптимизации циклов:

Варианты ответа:

- A) **сокращение количества повторений выполнения тела цикла;**
- B) сокращение тела цикла;
- C) представление циклов в виде блок-схем;
- D) трассировка циклов;
- E) поиск ошибок в циклах.

125. В чем сущность модульного программирования:

Варианты ответа:

- A) **в разбиении программы на отдельные функционально независимые части;**
- B) в разбиении программы на отдельные равные части;
- C) в разбиение программы на процедуры и функции;

126. Можно ли сочетать модульное и структурное программирование:

Варианты ответа:

- A) да;
- B) нет.

127. Может ли модуль включать несколько процедур или функций:

Варианты ответа:

- A) да;
- B) нет.

128. Рекомендуемые размеры модулей:

Варианты ответа:

- A) небольшие;**
- В) большие;
- С) равные;
- Д) фиксированной длины.

129. В чем заключается независимость модуля:

Варианты ответа:

- A) в написании, отладке и тестировании независимо от остальных модулей;**
- В) в разработке и написании независимо от других модулей;
- С) в независимости от работы основной программы.

130. При модульном программировании желательно, чтобы модуль имел:

Варианты ответа:

- А) большой размер;
- В) небольшой размер;**
- С) фиксированный размер;
- Д) любой размер.

131. Модульное программирование это:

Варианты ответа:

- A) разбиение программы на отдельные части;**
- В) структурирование;
- С) использование стандартных процедур и функций.

132. Можно ли использовать оператор GO TO в модульных программах:

Варианты ответа:

- A) можно;**
- В) нельзя.

133. Разрешается ли использование циклов при модульном программировании:

Варианты ответа:

- A) да;**
- В) нет.

134. Разрешается ли использование условных операторов при модульном программировании:

Варианты ответа:

- A) да;**
- В) нет.

135. Сократится ли размер программы, если ее написать в виде набора модулей:

Варианты ответа:

- A) нет;**
- В) да.

136. Достоинство модульного программирования:

Варианты ответа:

- A) создание программы по частям в произвольном порядке;**
- В) не требует компоновки;
- С) всегда дает эффективные программы;
- Д) снижает количество ошибок.

137. Недостаток модульного программирования:

Варианты ответа:

- А) увеличивает трудоемкость программирования;
- В) усложняет процедуру комплексного тестирования;**

- C) снижает быстроедействие программы;
- D) не позволяет выполнять оптимизацию программы.

138. Достоинство модульного программирования:

Варианты ответа:

- A) возможность приступить к тестированию до завершения написания всей программы;**
- B) не требует комплексного тестирования;
- C) уменьшает размер программы;
- D) повышает надежность программы.

139. Допустимо ли использование оператора GO TO при структурном программировании:

1. Варианты ответа:

- A) нет;**
- B) да.

140. Можно ли сочетать структурное программирование с модульным:

Варианты ответа:

- A) можно;**
- B) нельзя;
- C) только в особых случаях.

141. Любую ли программу можно привести к структурированному виду:

Варианты ответа:

- A) любую;**
- B) не все;
- C) нельзя.

142. Можно ли использовать оператор GO TO в структурированных программах:

Варианты ответа:

- A) можно;
- B) нельзя;**
- C) только в особых случаях.

143. Возможно, ли преобразовать неструктурированную программу к структурному виду:

Варианты ответа:

- A) да;**
- B) нет.

144. Возможно ли программирование без оператора GO TO:

Варианты ответа:

- A) да;**
- B) нет.

145. При структурном программировании задача выполняется:

Варианты ответа:

- A) поэтапным разбиением на более легкие задачи;**
- B) без участия программиста;
- C) объединением отдельных модулей программы.

146. Разрешается ли использование оператора GO TO при структурном программировании:

Варианты ответа:

- A) нет;**
- B) да;
- C) иногда.

147. Разрешается ли использование циклов при структурном программировании:

Варианты ответа:

A) да;

B) нет.

148. Разрешается ли использование оператора IF при структурном программировании:

Варианты ответа:

A) да;

B) нет.

149. Программирование без GO TO применяется. при:

Варианты ответа:

A) структурном программировании;

B) модульном программировании;

C) объектно-ориентированном программировании;

D) все ответы верные.

150. Достоинство структурного программирования:

Варианты ответа:

A) можно приступить к комплексному тестированию на раннем этапе разработки;

B) можно приступить к автономному тестированию на раннем этапе разработки;

C) нет необходимости выполнять тестирование;

D) можно пренебречь отладкой.

151. Достоинство структурного программирования:

Варианты ответа:

A) облегчает работу над большими и сложными проектами;

B) повышает быстродействие программы;

C) снижает затраты на программирование.

152. Недостаток структурного программирования:

Варианты ответа:

A) увеличивает размер программы;

B) снижает эффективность;

C) уменьшает количество ошибок;

D) не требует отладки.

153. Повышает ли читабельность программ структурное кодирование:

Варианты ответа:

A) да;

B) нет.

154. Разрешается ли использование циклов при объектно-ориентированном программировании:

Варианты ответа:

A) да;

B) нет.

2. 155. Разрешается ли использование оператора IF при объектно-ориентированном программировании:

3. Варианты ответа:

A) да;

B) нет.

156. Предусматривает ли объектно-ориентированное программирование использование стандартных процедур и функций:

A) да;

B) нет.

157. Можно ли сочетать объектно-ориентированное и структурное программирование

Варианты ответа:

A) **можно;**

B) нельзя.

158) Можно ли сочетать объектно-ориентированное и модульное программирование:

Варианты ответа:

A) **можно;**

B) нельзя.

159. Что такое объект, в объектно-ориентированном программировании:

Варианты ответа:

A) **тип данных;**

B) структура данных;

C) событие;

D) обработка событий;

E) использование стандартных процедур.

160. Инкапсуляция это:

Варианты ответа:

A) определение новых типов данных;

B) определение новых структур данных;

C) **объединение переменных, процедур и функций в одно целое;**

D) разделение переменных, процедур и функций;

E) применение стандартных процедур и функций.

161. Наследование это:

Варианты ответа:

A) передача свойств экземплярам;

B) передача свойств предкам;

C) **передача свойств потомкам;**

D) передача событий потомкам.

162. Полиморфизм это:

Варианты ответа:

A) **изменение поведения потомков, имеющих общих предков;**

B) передача свойств по наследству;

C) изменение поведения потомков на разные события;

D) изменение поведения экземпляров, имеющих общих предков;

163. Три "кита" объектно-ориентированного метода программирования:

Варианты ответа:

A) предки, родители, потомки;

B) **полиморфизм, инкапсуляция, наследование;**

C) свойства, события, методы;

D) визуальные, не визуальные компоненты и запросы.

164. Какое утверждение верно:

Варианты ответа:

A) предки наследуют свойства родителей;

B) родители наследуют свойства потомков;

C) потомки не могут иметь общих предков;

D) **потомки наследуют свойства родителей.**

165. Может ли дочерний элемент иметь двух родителей:

Варианты ответа:

- A) да;
- B) нет;**
- C) только для визуальных элементов;
- D) если их свойства совпадают.

165. Могут ли два визуальных компонента иметь общего предка:

Варианты ответа:

- A) да;**
- B) нет;
- C) если их свойства совпадают;
- D) если их методы совпадают.

167. Есть ли различие между объектом и экземпляром:

Варианты ответа:

- A) да;**
- B) нет;
- C) если у них общий предок.

168. Есть ли различие в поведении объекта и экземпляра того же типа:

Варианты ответа:

- A) да;
- B) если у них есть общий предок;
- C) нет;**
- D) если у них нет общего предков.

169. Изменение свойств, приводит к изменению поведения экземпляра:

Варианты ответа:

- A) нет;
- B) только для визуальных;
- C) только НЕ для визуальных ;
- D) да .**

170. Можно ли свойствам присваивать значения:

Варианты ответа:

- A) да (всегда);
- B) не всегда;**
- C) нет.

171. Можно ли переопределять методы:

Варианты ответа:

- A) да;**
- B) нет.

172. Можно ли переопределять свойства:

Варианты ответа:

- A) да;
- B) нет.**

173. Могут ли два различных объекта реагировать на событие по-разному:

Варианты ответа:

- A) да;**

В) нет.

174. Могут ли два экземпляра одного объекта реагировать на событие по-разному:

Варианты ответа:

А) да;

В) нет.

175. Какой методикой проектирования пользуются при структурном программировании:

Варианты ответа:

А) сверху вниз;

А) снизу-вверх.

176. Какой этап проектирования может быть исключен:

Варианты ответа:

А) эскизный проект;

В) технический проект;

С) рабочий проект.

177. Какие этапы проектирования можно объединять:

Варианты ответа:

А) технический и рабочий;

В) эскизный и рабочий;

С) технический и эскизный.

178. Модульное программирование применимо при:

Варианты ответа:

А) проектировании сверху вниз;

В) проектирование снизу-вверх;

179. Процесс преобразования постановки задачи в план алгоритмического или вычислительного решения это:

Варианты ответа:

А) проектирование;

В) анализ требований;

С) программирование;

Д) тестирование.

180. Составление спецификаций это:

Варианты ответа:

А) формализация задачи;

В) эскизный проект;

С) поиск алгоритма;

Д) отладка.

181. Этап разработки программы, на котором дается характеристика области применения программы:

Варианты ответа:

А) техническое задание;

В) эскизный проект;

С) технический проект;

Д) внедрение;

Е) рабочий проект.

182. Укажите правильную последовательность создания программы:

Варианты ответа:

А) формулирование задачи, анализ требований, проектирование, программирование;

В) анализ требований, проектирование, программирование, тестирование, отладка;

- C) анализ требований, программирование, проектирование, тестирование;
- D) анализ требований, проектирование, программирование, модификация, трассировка;
- E) формулирование задачи, анализ требований, программирование, проектирование, отладка.

183. Уточнение структуры входных и выходных данных, разработка алгоритмов, определение элементов интерфейса входят в:

Варианты ответа:

- A) технический проект;**
- B) рабочий проект;
- C) эскизный проект.

184. Несуществующий метод проектирования:

Варианты ответа:

- A) алгоритмическое;**
- B) нисходящее;
- C) восходящее.

185. Метод проектирования:

Варианты ответа:

- A) нисходящее;**
- B) алгоритмическое;
- C) логическое;
- D) использование языков программирования;
- E) составление блок-схем.

186. Нисходящее проектирование это:

Варианты ответа:

- A) последовательное уточнение (детализация);**
- B) составление блок-схем;
- C) разделение программы на отдельные участки (блоки);
- D) трассировка.

187. Признаки нисходящего программирования:

Варианты ответа:

- A) последовательная детализация;**
- B) наличие оптимизации;
- C) наличие тестирования;
- D) автоматизация программирования.

188. Какой методикой пользуются при структурном программировании:

Варианты ответа:

- A) сверху вниз;**
- B) снизу-вверх.

189. Проектирование сверху вниз это:

Варианты ответа:

- A) последовательное разбиение общих задач на более мелкие;**
- B) составление из отдельных модулей большой программы.

190. Проектирование снизу-вверх это:

Варианты ответа:

- A) составление из отдельных модулей большой программы;**
- B) последовательное разбиение общих задач на более мелкие.

191. Модульное программирование применимо при:

Варианты ответа:

- A) проектировании сверху вниз;
- B) проектирование снизу-вверх;
- C) и в том, и другом случае;**
- D) ни в коем случае.

192. Какой методикой проектирования пользуются при структурном программировании:

Варианты ответа:

A) сверху вниз;

B) снизу-вверх.

193. В чем заключается иерархический подход в решении задачи:

Варианты ответа:

A) в последовательном разбиении задачи на более мелкие составные части;

B) в выделении основных и второстепенных элементов;

C) в возможности параллельного выполнения отдельных частей задачи.

4. 194. Какой метод проектирования соответствует иерархическому подходу в решении задачи:

5. Варианты ответа:

A) нисходящее (сверху вниз);

B) восходящее (снизу-вверх).

195. В каких единицах измеряются затраты на проектирование:

Варианты ответа:

A) в человеко-днях;

B) в долларах;

C) в тенге;

D) в килобайтах.

196. Зависит ли трудоемкость разработки от сложности алгоритма:

Варианты ответа:

A) да;

B) нет.

197. Зависит ли трудоемкость разработки от количества программистов:

Варианты ответа:

A) да;

B) нет.

198. Зависит ли трудоемкость разработки от языка или системы программирования:

Варианты ответа:

A) да;

B) нет.

199. Зависит ли трудоемкость разработки от количества обрабатываемой информации:

Варианты ответа:

A) да;

B) нет.

200. Зависит ли трудоемкость разработки от вида информации:

Варианты ответа:

A) да;

B) нет.

201. Если вы приобрели программу законным путем, являетесь ли вы собственником программы:

Варианты ответа:

A) нет;

B) да.

202. Если вы приобрели программы законным путем, имеете ли вы право вносить в нее изменения:

Варианты ответа:

A) нет;

B) да

203. Если вы приобрели программы законным путем, имеете ли вы право продать ее:

Варианты ответа:

A) да;

B) нет.

204. Кому принадлежит право собственности на ПО:

Варианты ответа:

A) разработчику;

A) продавцу;

B) покупателю.

205. Кому принадлежит авторское право на ПО:

Варианты ответа:

1) разработчику;

2) продавцу;

3) покупателю.

206. Что охраняется законом:

Варианты ответа:

1) структура базы данных;

2) содержание базы данны

Ключ к тесту

1. 1	27. 1	53. 1
2. 1	28. 1	54. 1
3. 234	29. 1	55. 1
4. 3	30. 1	56. 1
5. 1	31. 4	57. 1
6. 1	32. 1	58. 1
7. 1	33. 1	59. 1
8. 1	34. 1	60. 1
9. 1	35. 1	61. 1
10. 1	36. 1	62. 1
11. 14	37. 1	63. 1
12. 1	38. 1	64. 1
13. 1	39. 1	65. 1
14. 1	40. 1	66. 1
15. 1	41. 1	67. 1
16. 1	42. 1	68. 1
17. 1	43. 1	69. 1
18. 1	44. 1	70. 1
19. 3	45. 1	71. 1
20. 1	46. 1	72. 1
21. 1	47. 1	73. 1
22. 1	48. 1	74. 1
23. 1	49. 1	75. 1
24. 1	50. 2	76. 1
25. 1	51. 1	77. 1
26. 1	52. 2	78. 1

79. 1		122.	A	165.	B
80. 1		123.	A	166.	A
81. 1		124.	A	167.	A
82. 1		125.	A	168.	C
83. 1		126.	A	169.	D
84. 1		127.	A	170.	B
85. 1		128.	A	171.	A
86. 1		129.	A	172.	B
87. 1		130.	B	173.	A
88. 1		131.	A	174.	A
89. 1		132.	A	175.	A
90. 1		133.	A	176.	A
91. 1		134.	A	177.	A
92. 1		135.	A	178.	B
93. 1		136.	A	179.	A
94. 1		137.	B	180.	A
95. 1		138.	A	181.	A
96. 1		139.	A	182.	A
97. 1		140.	A	183.	A
98. 1		141.	A	184.	A
99. 1		142.	B	185.	A
100.	4	143.	A	186.	A
101.	1	144.	A	187.	A
102.	3	145.	A	188.	A
103.	1	146.	A	189.	A
104.	2	147.	A	190.	A
105.	2	148.	A	191.	C
106.	A	149.	A	192.	A
107.	B	150.	A	193.	A
108.	A	151.	A	194.	A
109.	A	152.	A	195.	A
110.	A	153.	A	196.	A
111.	A	154.	A	197.	B
112.	A	155.	A	198.	A
113.	A	156.	A	199.	B
114.	A	157.	A	200.	A
115.	A	158.	A	201.	A
116.	A	159.	A	202.	A
117.	A	160.	C	203.	A
118.	A	161.	C	204.	A
119.	A	162.	A	205.	1
120.	A	163.	B	206.	2
121.	A	164.	D		

Критерии оценивания работ:

90-100% выполненной работы (34-38 заданий) – «5»

75-85% выполненной работы (28-33 заданий) – «4»

50-70% выполненной работы (19-27 заданий) – «3»

4.4 Промежуточная аттестация по МДК 01.03

Тест на знание основ разработки приложений для ОС Android

1. Ядро какой операционной системы использовалось в качестве базы для ОС Android?
(Отметьте один правильный вариант ответа.)

Вариант 1 Linux

Вариант 2 OS/2

Вариант 3 Mac OS

Вариант 4 Windows

2. Какой движок баз данных используется в ОС Android?
(Отметьте один правильный вариант ответа.)

Вариант 1 DBM

Вариант 2 SQLite

Вариант 3 InnoDB

Вариант 4 MyISAM

3. Какой компонент архитектуры Android позволяет любому приложению использовать уже реализованные возможности других приложений, к которым разрешен доступ?
(Отметьте один правильный вариант ответа.)

Вариант 1 Application Framework

Вариант 2 Linux Kernel

Вариант 3 Applications

Вариант 4 Libraries & Android Runtime

4. Инструмент разработки, позволяющий адаптировать код C/C++ для работы на Android, это -

(Отметьте один правильный вариант ответа.)

Вариант 1 Google Android SDK (ADT Bundle)

Вариант 2 Marmalade SDK

Вариант 3 Intel* Software Manager

Вариант 4 Android NDK

5. Чем являются Eclipse и IDEA?

(Отметьте один правильный вариант ответа.)

Вариант 1 API (интерфейс прикладного программирования)

Вариант 2 IDE (интегрированная среда разработки)

Вариант 3 SDK (набор средств разработки)

Вариант 4 ADT (инструменты разработки под Android)

6. К проблемам разработки под ОС Android можно отнести:

(Отметьте один правильный вариант ответа.)

Вариант 1 отсутствие эффективных инструментов разработки

Вариант 2 ненадежную изоляцию ядра системы от выполняемых приложений

Вариант 3 все варианты ответа верны

Вариант 4 большое разнообразие устройств, невозможность проверки приложения на всех

7. К преимуществам среды разработки Intel XDK можно отнести:

(Отметьте один правильный вариант ответа.)

Вариант 1 легкость разработки кроссплатформенных приложений

Вариант 2 все варианты ответа верны

Вариант 3 наличие облачного хранилища для разработанных приложений

Вариант 4 наличие собственного эффективного эмулятора

8. Среда разработки Intel XDK поддерживает следующие языки программирования:
(Отметьте один правильный вариант ответа.)

Вариант 1 C++

Вариант 2 C#

Вариант 3 Java

Вариант 4 JavaScript

9. Для запуска приложений, разработанных в Android IDE, необходимо:
(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 настроить среду и запустить проект на устройстве

Вариант 2 запустить виртуальную машину с установленной ОС Android

Вариант 3 настроить компьютер (для Windows необходимо установить нужный драйвер вручную, нужны права администратора)

Вариант 4 настроить устройство (включить режим отладки по USB)

10. Не является средством разработки под Android:
(Отметьте один правильный вариант ответа.)

Вариант 1 ProGuard

Вариант 2 Device Monitor

Вариант 3 AVD Manager

Вариант 4 SDK Manager

11. С какой целью был создан Open Handset Alliance?
(Отметьте один правильный вариант ответа.)

Вариант 1 писать историю развития ОС Android

Вариант 2 продавать смартфоны под управлением Android

Вариант 3 разрабатывать открытые стандарты для мобильных устройств

Вариант 4 рекламировать смартфоны под управлением Android

12. Какое название получила версия Android 4.4?
(Отметьте один правильный вариант ответа.)

Вариант 1 Krispy Kreme

Вариант 2 kream-karamel

Вариант 3 kitekat

Вариант 4 Kit Kat

13. С какой целью инструмент Intel* Hardware Accelerated Execution Manager (Intel* HAXM) используется в среде разработки Intel* Beacon Mountain?
(Отметьте один правильный вариант ответа.)

Вариант 1 для ускорения работы эмулятора в среде разработки

Вариант 2 для эффективного распараллеливания C++ мобильных приложений

Вариант 3 для оптимизированной обработки данных и изображений

Вариант 4 для оптимизации загрузки системы при использовании процедур OpenGL

14. С какой целью инструмент Intel* Graphics Performance Analyzers (Intel* GPA) System Analyzer используется в среде разработки Intel* Beacon Mountain?
(Отметьте один правильный вариант ответа.)

Вариант 1 позволить разработчикам эффективно распараллелить C++ мобильные приложения

Вариант 2 позволить разработчикам оптимизировать загрузенность системы при использовании процедур OpenGL

Вариант 3 для ускорения работы эмулятора в среде разработки

Вариант 4 для оптимизированной обработки данных и изображений

15. Тестирование на реальных устройствах приложений, разработанных в Intel XDK
(Отметьте один правильный вариант ответа.)

Вариант 1 невозможно

Вариант 2 можно проводить, установив на устройство специальную программу или собрав проект и установив результат сборки на устройство

Вариант 3 можно проводить, установив на устройство специальную программу

Вариант 4 можно проводить, собрав проект и установив результат сборки на устройство

16. Эмулятор среды разработки Intel XDK:

(Отметьте один правильный вариант ответа.)

Вариант 1 не требует больших системных ресурсов

Вариант 2 позволяет проверить работу приложений, использующих акселерометр

Вариант 3 позволяет моделировать исполнение приложения на ряде устройств

Вариант 4 все варианты ответа верны

17. Intel XDK поддерживает разработку под:

(Отметьте один правильный вариант ответа.)

Вариант 1 JavaFX Mobile

Вариант 2 Apple iOS, BlackBerry OS

Вариант 3 Android, Apple iOS, Microsoft Windows 8, Tizen

Вариант 4 MtkOS, Symbian OS, Microsoft Windows 8

18. Какая графическая библиотека входит в набор библиотек ОС Android?

(Отметьте один правильный вариант ответа.)

Вариант 1 OpenCV

Вариант 2 DirectX

Вариант 3 Open GL

Вариант 4 OpenCL

19. Приложения, не имеющие GUI и выполняющиеся в фоновом режиме - это

(Отметьте один правильный вариант ответа.)

Вариант 1 Intents

Вариант 2 Content Providers

Вариант 3 Activities

Вариант 4 Services

20. Удобное средство обмена между двумя NFC-устройствами:

(Отметьте один правильный вариант ответа.)

Вариант 1 Wi-Fi Direct

Вариант 2 Dalvik

Вариант 3 AndroidBeam

Вариант 4 Bluetooth

21. Преимуществом эмуляторов является:

(Отметьте один правильный вариант ответа.)

Вариант 1 невозможность полноценной отладки

Вариант 2 необходимость дополнительных системных ресурсов

Вариант 3 низкая стоимость

Вариант 4 медленный запуск

22. Фоновые приложения ...

(Отметьте один правильный вариант ответа.)

Вариант 1 после настройки не предполагают взаимодействия с пользователем, большую часть времени находятся и работают в скрытом состоянии

Вариант 2 небольшие приложения, отображаемые в виде графического объекта на рабочем столе

Вариант 3 выполняют свои функции и когда видимы на экране, и когда скрыты другими приложениями

Вариант 4 большую часть времени работают в фоновом режиме, однако допускают взаимодействие с пользователем и после настройки

23. Какая папка в структуре Android-приложения содержит файлы с исходным кодом на языке Java?

(Отметьте один правильный вариант ответа.)

Вариант 1 gen

Вариант 2 src

Вариант 3 bin

Вариант 4 res

24. Приложение какого вида имеет смысл использовать для отображения динамической информации, такой как заряд батареи, прогноз погоды, дата и время?

(Отметьте один правильный вариант ответа.)

Вариант 1 фоновое приложение

Вариант 2 приложение переднего плана

Вариант 3 виджет

Вариант 4 смешанное приложение

25. Что находится в папке gen?

(Отметьте один правильный вариант ответа.)

Вариант 1 ресурсы приложения

Вариант 2 аудио файлы

Вариант 3 изображения

Вариант 4 автоматически сгенерированные java-файлы

26. Каждый приемник широковещательных сообщений является наследником класса ...

(Отметьте один правильный вариант ответа.)

Вариант 1 ContentProvider

Вариант 2 BroadcastReceiver

Вариант 3 ViewReceiver

Вариант 4 IntentReceiver

27. Выберите верные утверждения относительно объекта-намерения (Intent).

(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 представляет собой структуру данных, содержащую описание операции, которая должна быть выполнена, и обычно используется для запуска активности или сервиса

Вариант 2 используются для передачи сообщений между основными компонентами приложений

Вариант 3 используется для получения инструкций от пользователя

Вариант 4 используется для передачи сообщений пользователю

28. Какие методы может использовать активность для запуска сервиса?

(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 bindService()

Вариант 2 openService()

Вариант 3 startService()

Вариант 4 createService()

29. Выберите верную последовательность действий, необходимых для создания в приложении контент-провайдера.

(Отметьте один правильный вариант ответа.)

Вариант 1 Проектирование способа хранения данных; Создание класса-наследника от класса ContentProvider; Определение строки авторизации провайдера, URI для его строк и имен столбцов

Вариант 2 Создание класса наследника от класса ContentProvider; Заполнение контент-провайдера данными; Определение способа работы с данными

Вариант 3 Проектирование способа хранения данных; Определение способа организации данных; Определение способа работы с данными

Вариант 4 Создание класса наследника от класса ContentProvider; Определение способа организации данных; Заполнение контент-провайдера данными

30. Какой метод вызывается системой в случае, когда активность теряет фокус?

(Отметьте один правильный вариант ответа.)

Вариант 1 onPause()

Вариант 2 onResume()

Вариант 3 onDestroy()

Вариант 4 onStop()

31. Какой вид компоновки определяет табличный способ расположения компонентов графического интерфейса пользователя в приложениях под Android?

(Отметьте один правильный вариант ответа.)

Вариант 1 TableLayout

Вариант 2 RowLayout

Вариант 3 GridBagLayout

Вариант 4 GridLayout

32. Дизайн или проектирование интерфейса для графических дизайнеров:

(Отметьте один правильный вариант ответа.)

Вариант 1 передача информации о поведении посредством ожидаемого назначения

Вариант 2 прозрачность и понятность информации

Вариант 3 тон, стиль, композиция, которые являются атрибутами бренда

Вариант 4 все варианты ответа верны

32. Следующие утверждения верны:

(Отметьте один правильный вариант ответа.)

Вариант 1 текстура бесполезна для передачи различий или привлечения внимания

Вариант 2 люди легко воспринимают контрастность

Вариант 3 восприятие направления затруднено при больших размерах объектов

Вариант 4 все варианты ответа верны

33. Элементы управления "Радиокнопки":

(Отметьте один правильный вариант ответа.)

Вариант 1 должны иметь квадратную форму

Вариант 2 должны иметь круглую форму

Вариант 3 не должны иметь круглую форму

Вариант 4 могут иметь любую форму

35. Какое средство организации интерфейса помогает визуальное уравновесить элементы?

(Отметьте один правильный вариант ответа.)

Вариант 1 сетка

Вариант 2 выравнивание

Вариант 3 логический маршрут

Вариант 4 симметрия

36. При создании справочной системы необходимо:
(Отметьте один правильный вариант ответа.)

Вариант 1 использовать всплывающие подсказки

Вариант 2 создать руководство по "быстрому старту"

Вариант 3 все вышеперечисленное

Вариант 4 учитывать, что она не должна становиться костылем для продукта

37. К элементам ввода относят:

(Отметьте один правильный вариант ответа.)

Вариант 1 ограничивающие элементы ввода

Вариант 2 ползунки

Вариант 3 счетчики

Вариант 4 все вышеперечисленное

38. Нижняя панель (Navigation Bar) предназначена

(Отметьте один правильный вариант ответа.)

Вариант 1 для вывода времени

Вариант 2 для вывода уровня заряда батареи

Вариант 3 для вывода уровня сигнала сотовой сети

Вариант 4 для навигации на тех устройствах, которые не имеют аппаратных навигационных клавиш

39. Один DP равен:

(Отметьте один правильный вариант ответа.)

Вариант 1 одному пикселю на экране типа MDPI

Вариант 2 0,5 дюйма

Вариант 3 300 dpi

Вариант 4 0,3 мм

40. Возможны следующие виды списков - элементов управления

(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 комбо-списки

Вариант 2 а и b

Вариант 3 раскрывающие списки

Вариант 4 нумерованные списки

41. Возможность прокрутки сеток стоит осуществлять

(Отметьте один правильный вариант ответа.)

Вариант 1 только по горизонтали

Вариант 2 по горизонтали и по вертикали

Вариант 3 по горизонтали или по вертикали

Вариант 4 только по вертикали

42. Текстовые гиперссылки при программировании для мобильных устройств:

(Отметьте один правильный вариант ответа.)

Вариант 1 стоит использовать в исключительных случаях

Вариант 2 стоит использовать для перехода между окнами приложения

Вариант 3 все варианты ответа неверны

Вариант 4 стоит использовать повсеместно

43. Какое визуальное свойство является самым главным в определении сущности объекта?

(Отметьте один правильный вариант ответа.)

- Вариант 1 размер
- Вариант 2 яркость
- Вариант 3 форма**
- Вариант 4 цвет

44. При проектировании окон приложения необходимо:

(Отметьте один правильный вариант ответа.)

Вариант 1 интегрировать в рабочее пространство инструменты, используемые наиболее часто

Вариант 2 для каждой функции делать отдельное диалоговое окно

Вариант 3 все вышеперечисленное

Вариант 4 позволять диалоговому окну оперировать только с одной функцией

45. Примерами комбо-элементов не являются:

(Отметьте один правильный вариант ответа.)

Вариант 1 комбо-список

Вариант 2 все вышеперечисленное

Вариант 3 комбо-поле

Вариант 4 комбо-кнопка

46. К традиционным типографическим инструментам не относят

(Отметьте один правильный вариант ответа.)

Вариант 1 цвет

Вариант 2 разреженность

Вариант 3 выравнивание по сетке

Вариант 4 масштаб

47. Минимальный размер элемента управления

(Отметьте один правильный вариант ответа.)

Вариант 1 64 dp

Вариант 2 48dp

Вариант 3 32px

Вариант 4 30pt

48. Выделяют следующие категории элементов управления:

(Отметьте один правильный вариант ответа.)

Вариант 1 системные элементы управления

Вариант 2 все варианты ответа верны

Вариант 3 элементы управления интерфейсами

Вариант 4 командные элементы управления

49. В чем заключается суть структурных шаблонов проектирования?

(Отметьте один правильный вариант ответа.)

Вариант 1 решают проблемы, относящиеся к взаимодействию с функциональными элементами

Вариант 2 решают проблемы, связанные с управлением отображением информации

Вариант 3 определяют тип продукта в отношении к пользователю

Вариант 4 все перечисленные

50. Какие элементы управления применяются для действий по настройке?

(Отметьте один правильный вариант ответа.)

Вариант 1 элементы ввода

Вариант 2 элементы отображения

Вариант 3 командные элементы управления

Вариант 4 элементы выбора

51. Следующие утверждения не верны:

(Отметьте один правильный вариант ответа.)

Вариант 1 на любом шаге должна быть возможность вернуться назад

Вариант 2 если объекты похожи, они должны выполнять сходные действия

Вариант 3 картинки работают быстрее, чем слова

Вариант 4 не используйте интерфейсные элементы

52. Расстояние между элементами управления рекомендуется делать кратным:

(Отметьте один правильный вариант ответа.)

Вариант 1 24dp

Вариант 2 8dp

Вариант 3 4dp

Вариант 4 16dp

53. Основные вкладки (FixedTabs) удобны при отображении

(Отметьте один правильный вариант ответа.)

Вариант 1 двух вкладок

Вариант 2 от четырех вкладок

Вариант 3 трех и менее вкладок

Вариант 4 трех и более вкладок

54. Объектами информационного дизайна являются

(Отметьте один правильный вариант ответа.)

Вариант 1 диаграммы

Вариант 2 все варианты ответа верны

Вариант 3 инфографика

Вариант 4 всевозможные графики

55. Расположение элементов мобильного приложения:

(Отметьте один правильный вариант ответа.)

Вариант 1 влияет на удобство использования

Вариант 2 полезно для передачи иерархии

Вариант 3 все варианты ответа верны

Вариант 4 полезно для создания пространственных отношений между объектами на экране и объектами реального мира

56. Использование ярлычков для всех панелей сетки и полупрозрачных "занавесок" для неактивных панелей

(Отметьте один правильный вариант ответа.)

Вариант 1 мешает пользователю понять, что скрывается под той или иной панелью

Вариант 2 скрывает от пользователя нужные панели

Вариант 3 нет правильного варианта ответа

Вариант 4 помогает пользователю быстрее понять, что скрывается под той или иной панелью

57. Строительные блоки визуального дизайна интерфейсов:

(Отметьте один правильный вариант ответа.)

Вариант 1 цвет

Вариант 2 все варианты ответа верны

Вариант 3 размер

Вариант 4 форма

58. Более крупные элементы:

(Отметьте один правильный вариант ответа.)

Вариант 1 привлекают больше внимания

Вариант 2 привлекают меньше внимания

Вариант 3 размер не влияет на уровень внимания

Вариант 4 все варианты ответа верны

59. К элементам управления отображением не относятся:

(Отметьте один правильный вариант ответа.)

Вариант 1 полосы прокрутки

Вариант 2 сетки и рамки

Вариант 3 разделители

Вариант 4 рукоятки

60. Выделяют следующие категории плотности экрана для Android-устройств:

(Отметьте один правильный вариант ответа.)

Вариант 1 правильный вариант ответа отсутствует

Вариант 2 HDPI, XHDPI, XXHDPI, и XXXHDPI

Вариант 3 LDPI, MDPI, HDPI

Вариант 4 LDPI, MDPI, HDPI, XHDPI, XXHDPI, и XXXHDPI

61. Всплывающие подсказки стоит использовать, когда
(Отметьте один правильный вариант ответа.)

Вариант 1 сообщение является важным и требует немедленного прочтения и ответа

Вариант 2 сообщение является важным, однако требует немедленного прочтения, но не ответа

Вариант 3 сообщение является важным, однако не требует немедленного прочтения и ответа

Вариант 4 сообщение не требует ответа пользователя, но важно для продолжения его работы

61. ProgressDialog это:

(Отметьте один правильный вариант ответа.)

Вариант 1 диалоговое окно, которое может содержать заголовок, до трех кнопок, список выбираемых значений или настраиваемое содержимое

Вариант 2 контейнер для создания собственных диалоговых окон

Вариант 3 диалоговое окно с предопределенным интерфейсом, позволяющее выбрать дату или время

Вариант 4 диалоговое окно, содержащее линейку процесса выполнения какого-то действия

62. DialogFragment это:

(Отметьте один правильный вариант ответа.)

Вариант 1 диалоговое окно, которое может содержать заголовок, до трех кнопок, список выбираемых значений или настраиваемое содержимое

Вариант 2 контейнер для создания собственных диалоговых окон

Вариант 3 диалоговое окно с предопределенным интерфейсом, позволяющее выбрать дату или время

Вариант 4 диалоговое окно, содержащее линейку процесса выполнения какого-то действия

63. При загрузке приложения первой появляется активность,

(Отметьте один правильный вариант ответа.)

Вариант 1 чье описание первым в лексикографическом порядке

Вариант 2 чье описание находится первым в манифесте

Вариант 3 чье описание находится последним в манифесте

Вариант 4 правильный вариант ответа отсутствует

64. Для чего предназначен класс ListActivity?

(Отметьте один правильный вариант ответа.)

Вариант 1 для создания нового листа

Вариант 2 для создания активности, основным элементом которой является список

Вариант 3 для создания обычной активности

Вариант 4 для создания активности, основным элементом которой является таблица

65. Интерфейс Adapter позволяет:

(Отметьте один правильный вариант ответа.)

Вариант 1 связывать список и названия его элементов

Вариант 2 оптимизировать процесс зарядки телефона

Вариант 3 обрабатывать нажатия

Вариант 4 адаптировать приложение к любой версии Android SDK

66. Для создания собственных диалоговых окон используют контейнер:

(Отметьте один правильный вариант ответа.)

Вариант 1 AlertDialog

Вариант 2 ProgressDialog

Вариант 3 DialogFragment

Вариант 4 DatePickerDialog

67. Можно ли создать диалоговое окно без управляющих кнопок?

(Отметьте один правильный вариант ответа.)

Вариант 1 нет, ни в коем случае

Вариант 2 да, если у него отсутствует содержательная часть

Вариант 3 нет, диалоговое окно должно содержать от одной до трех кнопок

Вариант 4 да, если элементы содержательной части являются кликабельными

68. Заголовок диалогового окна:

(Отметьте один правильный вариант ответа.)

Вариант 1 генерируется автоматически

Вариант 2 является обязательным элементом

Вариант 3 является необязательным элементом

Вариант 4 является необязательным элементом, если отсутствуют содержательная часть и управляющие кнопки

69. Шаблон Fixed Tabs + Swipe позволяет переключаться между вкладками:

(Отметьте один правильный вариант ответа.)

Вариант 1 только с помощью кнопок

Вариант 2 только с помощью перелистывания

Вариант 3 как с помощью кнопок, так и с помощью перелистывания

Вариант 4 не позволяет переключаться между вкладками

Ключ к тесту:

1. Вариант 1

2. Вариант 2

3. Вариант 1

4. Вариант 4

5. Вариант 2

6. Вариант 4

7. Вариант 2

8. Вариант 4

9. Вариант 1, Вариант 3,

Вариант 4

10. Вариант 1

11. Вариант 3

12. Вариант 4

13. Вариант 1

14. Вариант 2

15.

16.

17. Вариант 3

18. Вариант 3

19. Вариант 4

20. Вариант 3

21. Вариант 3

22. Вариант 1

23. Вариант 2

24. Вариант 3

25. Вариант 4

26. Вариант 2
27. Вариант 1, Вариант 2
28. Вариант 1, Вариант 3
29. Вариант 1
30. Вариант 1
31. Вариант 1
32. вариант 4
32. Вариант 4
33. Вариант 3
34. Вариант 2
35.
36. Вариант 3
37. Вариант 4
38. Вариант 4
39. Вариант 1
40.

41. Вариант 4
42. Вариант 1
43. Вариант 3
44. Вариант 1
45. Вариант 3
46. Вариант 1
47. Вариант 2
48. Вариант 4
49. Вариант 2
50. Вариант 4
51. Вариант 4
52. Вариант 2
53. Вариант 3
54. вариант 2
55. Вариант 3
56. Вариант 4

57. Вариант 2
58. Вариант 1
59. Вариант 4
60. Вариант 4
61. Вариант 4
61.Вариант 4
62. Вариант 2
63. Вариант 2
64. Вариант 2
65. Вариант 1
66. Вариант 3
67. Вариант 4
68. Вариант 4
69. Вариант 3

Критерии оценивания работ:

90-100% выполненной работы (34-38 заданий) – «5»

75-85% выполненной работы (28-33 заданий) – «4»

50-70% выполненной работы (19-27 заданий) – «3»

Промежуточная аттестация по МДК 01.04

Тест

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 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
 - 1100
 - 1001
18. Результат деления в двоичной системе счисления числа 110 на 11 равен
- 11
 - 10**
 - 01
 - 101
19. Перевести смешанное двоичное $111100000,101$ 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
- определить байт;
 - определить слово (2 байта);**
 - определить двойное слово (4 байта);
 - определить 6 байт
 - определить учетверенное слово (8 байт);
31. Псевдокоманда DD означает
- определить байт;
 - определить слово (2 байта);
 - определить двойное слово (4 байта);**
 - определить 6 байт
 - определить учетверенное слово (8 байт);
32. К сегментным регистрам относят
- CS, DS, SS и ES.**
 - AX, BX, CX и DX
 - SP и BP.
 - SI и DI
33. К индексным регистрам относят
- CS, DS, SS и ES.
 - AX, BX, CX и DX
 - SP и BP.
 - SI и DI**
34. Устанавливает бит результата в 1, если оба бита, бит источника и бит приемника установлены в 1.
- AND
 - OR**
 - XOR (НЕ ИЛИ)
 - NOT
35. устанавливает бит результата в 1, если бит источника отличается от бита приемника.
- AND**
 - OR
 - XOR
 - NOT
36. Запись регистра в память осуществляется при помощи команды
- MOV**
 - XCHG
 - XLAT
37. Порядковый номер элемента в массиве, который всегда начинается с нуля
- Индекс**
 - Команда
 - Псевдокоманда
 - Макрокоманда
38. Позволяет сохранять длину исходного текста программы
- Индекс
 - Команда
 - Псевдокоманда
 - Макрокоманда**

Ключ к тесту:

1.a	5.d	9.c
2.b	6.a	10.a
3.b	7.a	11.d
4.a	8.b	12.b

13.b	22.c	31.c
14.b	23.c	32.a
15.c	24.b	33.d
16.b	25.b	34.b
17.a	26.a	35.a
18.b	27.c	36.a
19.b	28.d	37.a
20.a	29.a	38.d
21.b	30.b	

Критерии оценивания работ:

90-100% выполненной работы (34-38 заданий) – «5»

75-85% выполненной работы (28-33 заданий) – «4»

50-70% выполненной работы (19-27 заданий) – «3»

Задачи

Для решения следующих задач разработать программы на Ассемблере.

- Поменять местами содержимое двух переменных ($X \leftrightarrow Y$).
- Выполнить циклический обмен содержимого трех переменных (X, Y, Z).
- Сохранить в переменных X, Y, Z , произведения числа, расположенного в заданной ячейке памяти, на числа 11, 12, 13, 14.
- Пусть в четырех последовательных ячейках памяти расположены числа (соответственно) $x, y, 10, 14$. Вычислить выражение: $10x - 14y$.
- Вычислить значения выражений:
 - $2x^2 + 3y^2 + 35xy - 18$;
 - $7x^3 + 4x + 5$;
 - $x^2 - 5x + 7$;
 - $ax^2 + bx + c$;
 - $x^4 - 15x + 20$.
- Для заданного X вычислите значение X^{32} , используя наименьшее количество команд и переменных.
- Вычислите значения выражений:

1) $ x^2 - 7x - 1 $;	2) $x^2 - 5x + 7 $.
-----------------------	-----------------------
- Вычислите значение выражений:

1) $y = \begin{cases} x - 5 , & \text{если } x \leq 0, \\ x - 5, & \text{если } x > 0; \end{cases}$	2) $y = \begin{cases} x + 5, & \text{если } x < 0, \\ x - 5, & \text{если } x \geq 0; \end{cases}$
1) $y = \begin{cases} x - 5, & \text{если } x \leq 0, \\ x - 5 , & \text{если } x > 0. \end{cases}$	
- Пусть задано некоторое целое положительное число X . С использованием конструкции цикла просуммировать все числа от 1 до X .
- Определить делится ли целое число X на целое число Y ($X > Y$), используя вычитание Y из X в цикле.
- Определить наибольший общий делитель (НОД(x, y)) двух натуральных чисел с помощью алгоритма Евклида, используя очевидное свойство: $\text{НОД}(x, y) = \text{НОД}(x - y, y)$, при $x > y$.
- Дано целое число X и натуральное N . Вычислить значение выражения X^N .
- Числа Фибоначчи определяются по формуле $U_{n+1} = U_n + U_{n-1}, U_1 = U_2 = 1$, т.е. каждый последующий член равен сумме двух предыдущих.
 - Определить значение числа Фибоначчи с заданным номером.
 - Найти сумму чисел Фибоначчи, не превышающих заданного K .
- Вычислить сумму числовой последовательности $1 + 3 + 5 + \dots + (2n - 1)$ и убедиться, что она равна n^2 .
- Вычислить сумму числовой последовательности $1 + 8 + 16 + \dots + 8(n - 1)$ и убедиться, что она равна $(2n - 1)^2$.

16. Отсортировать заданную последовательность чисел в возрастающем (убывающем) порядке.
17. Даны два отсортированных массива, объединить их в один, тоже отсортированный.
18. Для заданной последовательности чисел x_1, x_2, \dots, x_n определить их сумму: $S = x_1 + x_2 + \dots + x_n$ и сумму квадратов: $D = x_1^2 + x_2^2 + \dots + x_n^2$.

Контрольные вопросы к экзамену по МДК.01.01.

1. Программы и программное обеспечение
2. Система счисления. Двоичные числа
3. Система счисления. Шестнадцатеричные числа
4. Алгоритм Банкара
5. Алгоритм Медника
6. Алгоритм замещения страниц
7. Исследование файловой системы ввода/вывода
8. Директивы определения данных
9. Директивы распределения памяти
10. Регистры
11. Архитектура микропроцессоров
12. Предложения языка Ассемблер
13. Операнды команд
14. Алгоритмы работы Ассемблеров
15. Описание данных в Ассемблер
16. Команды пересылки общего назначения
17. Команды загрузки адресных значений и обращения к стеку
18. Команды ввода/вывода
19. Арифметические операции над двоичными числами. Сложение (вычитание) двоичных чисел без знака
20. Арифметические операции над двоичными числами. Сложение (вычитание) двоичных чисел со знаком
21. Арифметические операции над двоичными числами. Умножение двоичных чисел
22. Арифметические операции над двоичными числами. Деление двоичных чисел
23. Арифметические операции над десятичными числами. Сложение десятичных чисел
24. Арифметические операции над десятичными числами. Умножение десятичных чисел
25. Арифметические операции над десятичными числами. Деление десятичных чисел
26. Логические команды. Команды логических операций.
27. Логические команды. Команды сравнения.
28. Логические команды. Команды сдвигов.
29. Организация подпрограмм. Передача параметров в подпрограммы.
30. Организация макрокоманд. Передача параметров в макрокоманды

Вопросы к экзамену по МДК 01.02

1. Тестирование программных продуктов. Цель, стадии тестирования, виды тестирования, подходы к формированию тестов.
2. Тестирование программных продуктов. Стоимость исправления ошибок, этапы тестирования, место стадии тестирования в жизненном цикле программного продукта.
3. Тестирование программных продуктов. Подходы к формированию тестов. Ручной контроль. Методы ручного контроля.
4. Тестирование программных продуктов. Подходы к формированию тестов. Структурное тестирование. Критерии формирования тестовых наборов при структурном тестировании.
5. Тестирование программных продуктов. Подходы к формированию тестов. Функциональное тестирование. Методы формирования тестовых наборов при функциональном тестировании.

6. Нисходящее и восходящее тестирование, критерии завершения тестирования, оценочное тестирование и его виды.
7. Отладка программного продукта, причины сложности отладки программного продукта. Классификация ошибок при отладке программного продукта. Общая методика отладки программного продукта.
8. Отладка программного продукта. Классификация ошибок при отладке программного продукта. Методы и средства получения дополнительной информации при отладке программного продукта. Общая методика отладки программного продукта.
9. Программная документация. Виды программных документов. Основные правила оформления текстовых документов.

Контрольные вопросы к экзамену по МДК 01.03

1. Мобильное программирование, платформы для разработки.
2. Система Windows Phone 7.
3. Microsoft Visual Studio Express for Windows Phone.
4. Аппаратные средства устройств, поддерживающих Windows Phone 7.
5. Инструменты разработки Windows Phone7-приложений. Пример простейших программ Windows Phone 7-приложения. Запуск приложения на эмуляторе.
6. История появления, аспекты применения Microsoft Silverlight.
7. Основы работы с сенсорным вводом. Обработка нескольких касаний. Использование изображений.
8. Акселерометр и служба определения местоположения, вторичные потоки выполнения, обработка асинхронных операций и доступ к Веб-сервисам.
9. Особенности использования pivot и panorama.
10. Принципы интерфейса системы и приложений Metro
11. Типографика.
12. Краткая история ОС Android.
13. Intel для Android: партнерство и инструментарий разработчика.
14. Архитектура приложений для Android. Ресурсы приложения. Пользовательский интерфейс. Инструментарий разработки приложений для Android.
15. Обзор шагов разработки типичного приложения под Android. Особенности разработки с использованием эмулятора. Отладка кода в эмуляторе и на реальных устройствах. Пример простейших программ Android-приложения. Запуск приложения на эмуляторе.
16. Тестирование приложения с помощью Dalvik Debug Monitor Server (DDMS).
17. Планирование кадровой анимации, анимирование, анимация шаблонов, видов, использование класса Camera.
18. Проверка безопасности, работа со службами, основанными на местоположении, использование HTTP-служб, службы AIDL.
19. Подготовка AndroidManifest.xml для загрузки, локализация приложения, подготовка ярлыка приложения, подготовка APK-файла для загрузки, работа пользователя с Android Market.
20. Intel Power Monitoring Tool. Intel Graphics Performance Analyzer. Intel Energy Checker SDK. Intel Hardware Accelerated Execution Manager.
21. РаботасинструментамиIntelдляоптимизацииотладкиAndroid-приложений.
22. Текстовые элементы управления, кнопки, списки, таблицы, управление датой и временем, MapView.
23. Галерея, счетчик, диспетчеры шаблонов, адаптеры.
24. Создание меню, расширенные меню, загрузка меню при помощи XML-файлов, создание диалоговых окон, диалоговые окна с подсказками и предупреждениями.

Контрольные вопросы к экзамену по МДК01.04.

1. Программы и программное обеспечение

2. Система счисления. Двоичные числа
3. Система счисления. Шестнадцатеричные числа
4. Алгоритм Банкира
5. Алгоритм Медника
6. Алгоритм замещения страниц
7. Исследование файловой системы ввода/вывода
8. Директивы определения данных
9. Директивы распределения памяти
10. Регистры
11. Архитектура микропроцессоров
12. Предложения языка Ассемблер
13. Операнды команд
14. Алгоритмы работы Ассемблеров
15. Описание данных в Ассемблер
16. Команды пересылки общего назначения
17. Команды загрузки адресных значений и обращения к стеку
18. Команды ввода/вывода
19. Арифметические операции над двоичными числами. Сложение (вычитание) двоичных чисел без знака
20. Арифметические операции над двоичными числами. Сложение (вычитание) двоичных чисел со знаком
21. Арифметические операции над двоичными числами. Умножение двоичных чисел
22. Арифметические операции над двоичными числами. Деление двоичных чисел
23. Арифметические операции над десятичными числами. Сложение десятичных чисел
24. Арифметические операции над десятичными числами. Умножение десятичных чисел
25. Арифметические операции над десятичными числами. Деление десятичных чисел
26. Логические команды. Команды логических операций.
27. Логические команды. Команды сравнения.
28. Логические команды. Команды сдвигов.
29. Организация подпрограмм. Передача параметров в подпрограммы.
30. Организация макрокоманд. Передача параметров в макрокоманды

Критерии оценки МДК 01.01, МДК 01.02, МДК 01.03, МДК 01.04

– «Отлично» - теоретическое содержание МДК освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

– «Хорошо» - теоретическое содержание МДК освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

– «Удовлетворительно» - теоретическое содержание МДК освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

– «Неудовлетворительно» - теоретическое содержание МДК не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

Информационное обеспечение обучения

Основная литература

1. Белугина, С. В. Разработка программных модулей программного обеспечения для компьютерных систем. Прикладное программирование : учебное пособие для СПО / С. В. Белугина. - 3-е изд., стер. - Санкт-Петербург : Лань, 2022. - 312 с. - URL: <https://e.lanbook.com/book/200390> (дата обращения: 28.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-8114-9817-8. – Текст : электронный.
2. Федорова, Г. Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности : учебное пособие / Г. Н. Федорова. - Москва : КУРС : ИНФРА-М, 2022 - 336 с. - (Среднее Профессиональное Образование). - URL: <https://znanium.com/catalog/product/1858587> (дата обращения: 28.03.2022). - Режим доступа: ЭБС Znanium.com, для зарегистрир. пользователей. - ISBN 978-5-906818-41-6. - Текст : электронный.
3. Гагарина, Л. Г. Технология разработки программного обеспечения : учебное пособие / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Сидорова-Виснадул ; под ред. Л. Г. Гагариной. - Москва : ФОРУМ : ИНФРА-М, 2021. - 400 с. - (Среднее профессиональное образование). - URL: <https://znanium.com/catalog/product/1189951> (дата обращения: 29.03.2022). - Режим доступа: ЭБС Znanium.com, для зарегистрир. пользователей. - ISBN 978-5-8199-0812-9. - Текст : электронный.
4. Зубкова, Т. М. Технология разработки программного обеспечения : учебное пособие / Т. М. Зубкова. - Санкт-Петербург : Лань, 2022. - 324 с. - URL: <https://ezpro.fa.ru:3178/book/206882> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-8114-3842-6. - Текст : электронный.
5. Волк, В. К. Практическое введение в программную инженерию : учебное пособие / В. К. Волк. - Санкт-Петербург : Лань, 2022. - 100 с. - URL: <https://ezpro.fa.ru:3178/book/206669> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-8114-3656-9. - Текст : электронный.
6. Белугина, С. В. Разработка программных модулей программного обеспечения для компьютерных систем. Прикладное программирование : учебное пособие для СПО / С. В. Белугина. - 3-е изд., стер. - Санкт-Петербург : Лань, 2022. - 312 с. - URL: <https://e.lanbook.com/book/200390> (дата обращения: 28.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-8114-9817-8. - Текст : электронный.
7. Коржинский С. Н. Разработка мобильных приложений : учебник / Коржинский С. Н. - Москва : КноРус, 2022. - 421 с. - URL: <https://book.ru/book/944559> (дата обращения: 18.08.2022). - Режим доступа: ЭБС Book.ru, для зарегистрир. пользователей. - ISBN 978-5-406-09570-6. - Текст : электронный.
8. Соколова, В. В. Разработка мобильных приложений : учебное пособие для среднего профессионального образования / В. В. Соколова. - Москва : Юрайт, 2022. - 175 с. - (Профессиональное образование). - URL: <https://ezpro.fa.ru:3217/bcode/495527> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Юрайт, для зарегистрир. пользователей. - ISBN 978-5-534-10680-0. - Текст : электронный.
9. Заяц, А. М. Введение в гибридные технологии разработки мобильных приложений : учебное пособие для СПО / А. М. Заяц, Н. П. Васильев. - 2-е изд., стер. - Санкт-Петербург : Лань, 2022. - 160 с. - URL: <https://e.lanbook.com/book/200459> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-8114-9555-9. - Текст : электронный.
10. Пирская, Л. В. Разработка мобильных приложений в среде Android Studio : учебное пособие / Л. В. Пирская ; Южный федеральный университет. - Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2019. - 125 с. : ил., табл. - URL: <https://biblioclub.ru/index.php?page=book&id=598634> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Университетская библиотека онлайн, для зарегистрир. пользователей. - Библиогр. в кн. - ISBN 978-5-9275-3346-6. - Текст : электронный.
11. Кузнецов, А. С. Системное программирование : учебное пособие / А. С. Кузнецов, И. А. Якимов, П. В. Пересунько. - Красноярск : СФУ, 2018. - 170 с. - URL: <https://e.lanbook.com/book/157574> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-7638-3885-5. - Текст : электронный.

12. Васильева, И. И. Системное и прикладное программирование : учебное пособие / И. И. Васильева. - Елец : ЕГУ им. И.А. Бунина, 2019. - 130 с. - URL: <https://e.lanbook.com/book/195791> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-00151-039-0. - Текст : электронный.

Электронные издания (электронные ресурсы)

1. Учебники по программированию <http://programm.ws/index.php>
2. Eckel B. Thinking in C++ (2nd Edition) Free Electronic Book.
3. <http://www.mindview.net/Books/TICPP/ThinkingInCPP2e.html>

Дополнительная литература

1. Подбельский, В. В. Программирование. Базовый курс C# : учебник для среднего профессионального образования / В. В. Подбельский. - Москва : Юрайт, 2022. - 369 с. - (Профессиональное образование). - URL: <https://ezpro.fa.ru:3217/bcode/475775> (дата обращения: 28.03.2022). - Режим доступа: ЭБС Юрайт, для зарегистрир. пользователей. - ISBN 978-5-534-11467-6. - Текст : электронный.

2. Гниденко, И. Г. Технология разработки программного обеспечения : учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. - Москва : Юрайт, 2022. - 235 с. - (Профессиональное образование). - URL: <https://ezpro.fa.ru:3217/bcode/492496> (дата обращения: 28.03.2022). - Режим доступа: ЭБС Юрайт, для зарегистрир. пользователей. - ISBN 978-5-534-05047-9. - Текст : электронный.

3. Исаченко, О. В. Программное обеспечение компьютерных сетей : учебное пособие / О. В. Исаченко. - 2-е изд., испр. и доп. - Москва : ИНФРА-М, 2021. - 158 с. - (Среднее профессиональное образование). - URL: <https://znanium.com/catalog/product/1189344> (дата обращения: 29.03.2022). - Режим доступа: ЭБС Znanium.com, для зарегистрир. пользователей. - ISBN 978-5-16-015447-3. - Текст : электронный.

4. Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для среднего профессионального образования / Е. А. Черткова. - 2-е изд., испр. и доп. - Москва : Юрайт, 2022. - 147 с. - (Профессиональное образование). - URL: <https://ezpro.fa.ru:3217/bcode/493226> (дата обращения: 28.03.2022). - Режим доступа: ЭБС Юрайт, для зарегистрир. пользователей. - ISBN 978-5-534-09823-5. - Текст : электронный.

5. Немцова, Т. И. Программирование на языке высокого уровня. Программирование на языке C++ : учебное пособие / Т. И. Немцова, С. Ю. Голова, А. И. Терентьев ; под ред. Л. Г. Гагариной. - Москва : ФОРУМ : ИНФРА-М, 2021. - 512 с. + Доп. Материалы. - (Среднее профессиональное образование). - URL: <https://znanium.com/catalog/product/1172261> (дата обращения: 29.03.2022). - Режим доступа: ЭБС Znanium.com, для зарегистрир. пользователей. - ISBN 978-5-8199-0699-6. - Текст : электронный.

6. Мартишин, С. А. Основы теории надежности информационных систем : учебное пособие / С. А. Мартишин, В. Л. Симонов, М. В. Храпченко. - Москва : ФОРУМ : ИНФРА-М, 2020. - 255 с. - (Высшее образование: Бакалавриат). - URL: <https://znanium.com/catalog/product/1062374> (дата обращения: 04.07.2022). - Режим доступа: ЭБС Znanium.com, для зарегистрир. пользователей. - ISBN 978-5-8199-0757-3. - Текст : электронный

7. Советов, Б. Я. Информационные технологии: теоретические основы : учебное пособие / Б. Я. Советов, В. В. Цехановский. - 2-е изд., стер. - Санкт-Петербург : Лань, 2021. - 444 с. - URL: <https://ezpro.fa.ru:3178/book/167404> (дата обращения: 28.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-8114-1912-8. - Текст : электронный.

8. Ехлаков, Ю. П. Управление программными проектами. Стандарты, модели : учебное пособие для вузов / Ю. П. Ехлаков. - 3-е изд., стер. - Санкт-Петербург : Лань, 2021. - 244 с. - URL: <https://ezpro.fa.ru:3178/book/175498> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-8114-8362-4. - Текст : электронный.

9. Беспалов, Д. А. Операционные системы реального времени и технологии разработки кроссплатформенного программного обеспечения : учебное пособие : в 3 частях / Д. А. Беспалов, С. М. Гушанский, Н. М. Коробейникова ; Южный федеральный университет. - Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2021. - Часть 3. - 214 с. : ил. -

URL: <https://biblioclub.ru/index.php?page=book&id=683905> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Университетская библиотека онлайн, для зарегистрир. пользователей. - Библиогр.: с. 187-188. - ISBN 978-5-9275-3628-3 (Ч. 3). - ISBN 978-5-9275-3366-4. - Текст : электронный.

10. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для среднего профессионального образования / А. Ф. Тузовский. - Москва : Юрайт, 2022. - 218 с. - (Профессиональное образование). - URL: <https://ezpro.fa.ru:3217/bcode/495109> (дата обращения: 29.03.2022). - Режим доступа: ЭБС Юрайт, для зарегистрир. пользователей. - Библиогр. в кн. - ISBN 978-5-534-10017-4. - Текст : электронный.

11. Соколова, В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений : учебное пособие для вузов / В. В. Соколова. - Москва : Юрайт, 2022. - 175 с. - (Высшее образование). - URL: <https://ezpro.fa.ru:3217/bcode/490305> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Юрайт, для зарегистрир. пользователей. - ISBN 978-5-9916-6525-4. - Текст : электронный.

12. Жулабова, Ф. Т. Системное программирование. Лабораторные работы : учебное пособие для спо / Ф. Т. Жулабова. - 2-е изд., стер. - Санкт-Петербург : Лань, 2021. - 208 с. - URL: <https://e.lanbook.com/book/164955> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-8114-7721-0. - Текст : электронный.

13. Гунько, А. В. Системное программирование в среде Linux : учебное пособие / А. В. Гунько. - Новосибирск : НГТУ, 2020. - 235 с. - URL: <https://e.lanbook.com/book/152228> (дата обращения: 09.03.2022). - Режим доступа: ЭБС Лань, для зарегистрир. пользователей. - ISBN 978-5-7782-4160-2. - Текст : электронный.

1.