

МИНОБРНАУКИ РОССИИ

Глазовский инженерно-экономический институт (филиал)
федерального государственного бюджетного образовательного
учреждения высшего образования
«Ижевский государственный технический университет имени М.Т.Калашникова»
(ГИЭИ (филиал) ФГБОУ ВО «ИжГТУ имени М.Т. Калашникова»)



УТВЕРЖДАЮ

Директор

М.А. Бабушкин

15 июня 2023 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине

МДК.01.03 «Разработка мобильных приложений»

09.02.07 Информационные системы и программирование

Фонд оценочных средств разработан на основе Федерального государственного образовательного стандарта по специальности среднего профессионального образования 09.02.07 "Информационные системы и программирование", утвержденного приказом Министерства образования и науки Российской Федерации 09 декабря 2016 г. № 1547 с изменениями и дополнениями (приказ Министерства просвещения Российской Федерации от 17.12.2020 № 747 «О внесении изменений в федеральные государственные образовательные стандарты среднего профессионального образования» (зарегистрирован 22.01.2021 № 62178), приказ Министерства просвещения Российской Федерации от 01.09.2022 № 796 «О внесении изменений в федеральные государственные образовательные стандарты среднего профессионального образования» (зарегистрирован 11.10.2022 № 70461)).


Организация: ГИЭИ (филиал) ФГБОУ ВО «ИжГТУ имени М.Т. Калашникова»
разработчик: Калашникова»
Разработчик: Горбушин Денис Шарибзянович, преподаватель СПО
Утверждено: Протокол Ученого совета филиала № 7, от 14 июня 2023 г.

Руководитель образовательной программы


_____ Т.А. Савельева

15 июня 2023 г.

Согласовано: Начальник отдела по учебно-методической работе


_____ И.Ф. Яковлева

15 июня 2023 г.

СОДЕРЖАНИЕ

| | |
|---|-----------|
| 1. ПАСПОРТ КОНТРОЛЬНО-ОЦЕНОЧНЫХ СРЕДСТВ | 4 |
| 2. ОСНОВНЫЕ ПОКАЗАТЕЛИ ОЦЕНКИ РЕЗУЛЬТАТОВ | 6 |
| 3. РАСПРЕДЕЛЕНИЕ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ВИДАМ КОНТРОЛЯ И ЭЛЕМЕНТАМ ЗНАНИЙ И УМЕНИЙ | 9 |
| 4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ АТТЕСТАЦИИ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ | 12 |
| 4.1 Вопросы для устного опроса по темам | 12 |
| 4.2 Тестовые задания | 12 |
| 4.3 Контрольно-оценочный материал для проверочных работ | 30 |
| 5.2 Контрольно-оценочный материал для экзамена, зачёта | 35 |
| 5. ПЕРЕЧЕНЬ УЧЕБНЫХ ИЗДАНИЙ | 42 |

СОДЕРЖАНИЕ

| | |
|--|----|
| 1 Назначение комплекса оценочных средств (КОС)..... | 6 |
| 2 Перечень основных показателей оценки результатов, элементов знаний и умений, подлежащих текущему контролю и промежуточной аттестации..... | 6 |
| 3 Распределение основных показателей оценки результатов по видам аттестации..... | 8 |
| 4 Содержательно-компетентностные матрицы оценочных средств..... | 9 |
| 5 Структура банка КОС для текущей контроля и промежуточной аттестации..... | 10 |

1 Назначение комплекса оценочных средств (КОС)

Комплекс оценочных средств (КОС) предназначен для контроля и оценки образовательных достижений обучающихся при освоении программы *МДК 01.04 «Системное программирование»*

КОС включает контрольные материалы для проведения текущего контроля и промежуточной аттестации в форме экзамена.

КОС разработаны на основании положений:

ФГОС СПО по специальности 09.02.07 Информационные системы и программирование программ подготовки специалистов среднего звена по специальности

09.02.07 Информационные системы и программирование рабочей программы *МДК 01.04 Системное программирование*

2 Перечень основных показателей оценки результатов, элементов знаний и умений, подлежащих текущему контролю и промежуточной аттестации

| Результаты (освоенные профессиональные компетенции) | Основные показатели оценки результата | Формы и методы контроля и оценки |
|--|--|---|
| ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. | Разрабатывают спецификаций компонентов в соответствии с правилами создания спецификации. | Дифференцированный зачет. |
| | Разрабатывают код программного продукта на основе готовой спецификации на уровне модуля в соответствии со справочной системой программной среды разработки. | Оценка защиты курсового проекта. |
| | Применяют инструментальные средства автоматизации оформления документации в соответствии с правилами справочной системы инструментальных средств разработки. | Наблюдение за действиями на практике. |

| | | |
|--|--|---|
| | <p>Тестируют программный продукт на уровне модулей в соответствии с методикой тестирования</p> | <p>Оценка защиты лабораторных работ.</p> |
| | <p>Составляют техническое задание для разрабатываемого программного продукта в соответствии с требованиями ГОСТ 19.201-78.</p> | <p>Тестовый контроль.</p> |
| <p>ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием.</p> | <p>Выбирают и применяют утилиты автоматизированного проектирования, отладки и тестирования программных модулей в соответствии с правилами справочной системы среды программирования.</p> | <p>Дифференцированный зачет.</p> <p>Наблюдение за действиями на практике.</p> |
| | <p>Разрабатывают код программного модуля на современных языках программирования в соответствии с правилами справочной системы используемой среды программирования.</p> | <p>Оценка защиты лабораторных работ.</p> |
| | <p>Разрабатывают программу по разработанному алгоритму как отдельного модуля в соответствии с правилами справочной системы используемой среды программирования.</p> | <p>Тестовый контроль.</p> |
| | <p>Модифицируют программные продукты в инструментальной среде в соответствии с правилами справочной системы среды программирования.</p> | |
| | <p>Выбирают инструментальные средства на этапе отладки программного продукта в соответствии с методикой отладки.</p> | |
| <p>ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств..</p> | <p>Выполняют отладку программного продукта в соответствии с методикой отладки.</p> | <p>Дифференцированный зачет.</p> |
| | <p>Выполняют отладку программ на уровне модулей в соответствии с методикой отладки.</p> | <p>Наблюдение за действиями на практике.</p> |
| | <p>Используют специализированный отладчик в соответствии с правилами справочной системы среды программирования.</p> | <p>Оценка защиты лабораторных работ.</p> |
| | <p>Применяют приемы программирования, встраивающие отладку в программный код в соответствии с правилами справочной системы среды программирования..</p> | |
| | <p>Выбирают принципы отладки и тестирования программных продуктов в соответствии с правилами справочной системы среды программирования.</p> | |

| | | |
|---|---|---|
| ПК 1.4. Выполнять тестирование программных модулей. | Выполняют отладку и тестирование программы на уровне модуля в соответствии с правилами справочной системы среды программирования. | Дифференцированный зачет. Наблюдение за действиями на практике. Оценка защиты лабораторных работ. Тестовый контроль. |
| | Проводят тестирование программного модуля по определенному сценарию в соответствии с правилами справочной системы среды программирования. | |
| | Обосновывать внесение изменений в программу для защиты информации в соответствии с методикой защиты. | |
| | Используют аппаратные средства защиты в соответствии с правилами их применения. | |
| | Используют программные средства защиты в соответствии с правилами их применения. | |
| | Обнаруживают и локализуют ошибки программного продукта в соответствии с правилами справочной системы среды программирования. | |
| ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода. | Оптимизируют программный код модуля в соответствии с правилами справочной системы среды программирования. | Дифференцированный зачет. Наблюдение за действиями на практике. Оценка защиты лабораторных работ. Тестовый контроль. |
| | Выбирают методы и средства разработки проектной и технической документации с использованием графических языков спецификации в соответствии с правилами справочной системы графических языков. | |

3 Распределение основных показателей оценки результатов по видам аттестации

| Код и наименование элемента умений или знаний | Виды аттестации | |
|--|------------------|--------------------------|
| | Текущий контроль | Промежуточная аттестация |
| У1 Осуществлять разработку кода программного модуля на современных языках программирования | + | |
| У2 Создавать программу по разработанному алгоритму как отдельный модуль | | + |
| У3 Выполнять отладку и тестирование программы на уровне модуля | + | |
| У4 Оформлять документацию на программные средства | + | + |
| У5 Использовать инструментальные средства для автоматизации оформления документации | + | + |
| 31 Основные этапы разработки программного обеспечения | + | + |
| 32 Основные принципы технологии структурного и объектно-ориентированного программирования | + | + |
| 33 Основные принципы отладки и тестирования программных продуктов | + | + |
| 34 Методы и средства разработки технической документации | + | + |

4 Содержательно-компетентностная матрица оценочных средств

| Содержание учебного материала по программе УД | Текущий контроль | | Промежуточная аттестация | |
|--|-----------------------------|-------------------------|------------------------------------|-------------------------|
| | Проверяемые У, З, ОК, ПК | Код оценочного средства | Проверяемые У, З, ОК | Код оценочного средства |
| Системное программирование | | | У 2,4,5, З 1,2,3,4 ПК1.1-1.6 | 9 |
| Раздел 1 Современные системы программирования | У1-5 З 1-4 ПК 1.1-1.2 | 10 | | |
| Раздел 2 Машинно-ориентированная система программирования Ассемблер. Использование транслятора Turbo Assembler при разработке программ | У1-5 З 1-4 ПК 1.3-1.4 | 10 | | |
| Раздел 3 Разработка, отладка и тестирование программ для многозадачных операционных системы (ОС) на примере ОС Windows | У1-5 З 1-4 ПК 1.5-1.6 | 10 | | |

Спецификация экзаменационного задания

Назначение

Спецификацией устанавливаются требования к содержанию и оформлению вариантов оценочного средства.

Экзаменационное задание входит в состав комплекса оценочных средств и предназначено для *промежуточной аттестации* и оценки знаний и умений аттестуемых, соответствующих основным показателям оценки результатов подготовки по программе 09.02.07 Информационные системы и программирование рабочей программы *МДК 01.04 Системное программирование*

Условия аттестации: *аттестация проводится в форме устного экзамена по завершении освоения учебного материала учебной дисциплины при положительных результатах текущего контроля.*

Время контроля:

подготовка 30 мин.;

оформление и сдача 15 мин.;

всего 45 мин.

План варианта экзаменационного задания (соотношение контрольных задач/вопросов с содержанием учебного материала в контексте характера действий аттестуемых)

| Содержание учебного материала по программе УД | Коды ОПОР / литера категории действия/ количество контрольных задач/вопросов в билете | | | Общее количество задач/вопросов по категориям действий | | | | |
|--|---|--------------|--------------|--|---|---|---|---|
| | ПК 1.1 – 1.2 | ПК 1.3 – 1.4 | ПК 1.5 – 1.6 | | | | | |
| | | | | В | П | А | С | О |
| Раздел 1 Современные системы программирования | П1 | | | | 1 | | | |
| Раздел 2 Машинно-ориентированная система программирования Ассемблер. Использование транслятора Turbo Assembler при разработке программ | П1 | С1 | | | 1 | | 1 | |
| Раздел 3 Разработка, отладка и тестирование программ для многозадачных операционных системы (ОС) на примере ОС Windows | | А1 | О1 | | | 1 | | 1 |
| Всего | 5 | | | | 2 | 1 | 1 | 1 |

Структура варианта экзаменационного задания

Основная задача: оценка знаний и умений аттестуемых, соответствующих основным показателям оценки результатов подготовки по программе учебной дисциплины

Краткая характеристика: Экзаменационное задание является комплексным, т.к. включает практическую и теоретическую части, объединенные логически. Теоретическая часть требует развернутого ответа, практическая часть выполняется на макете или с использованием прикладных средств моделирования.

Экзаменационный билет

- 1) *Теоретический вопрос*
- 2) *Разработка программы на Assembler*
- 3) *Разработка программы на Borland C++ Builder*

7 Система оценки решения задач, ответов на вопросы, выполнения заданий

| % результативности (правильных ответов) | Качественная оценка уровня подготовки | |
|--|---------------------------------------|---------------------|
| | Отметка | Вербальный аналог |
| 90 ÷ 100 | 5 | отлично |
| 80 ÷ 89 | 4 | хорошо |
| 70 ÷ 79 | 3 | удовлетворительно |
| менее 70 | 2 | неудовлетворительно |

Трудоемкость

| Трудоемкость выполнения/решения, мин (час) | Количество задач | | |
|--|------------------|----|----|
| | 1 | 2 | 3 |
| Одной (го) задачи/вопроса | 10 | 10 | 10 |
| | 30 мин. | | |

Рекомендуемая литература для разработки оценочных средств и подготовки обучающихся к аттестации

Основные источники:

1. Канцедал С.А. Программирование на языках высокого уровня М.: ИТ Форум: ИНФРА – М, 2017
2. Партыка Т. Л., Попов И. И. Операционные системы, среды и оболочки. 2-е издание. Стереотип М.: Форум, 2015

Дополнительные источники:

1. Гордеев А.В., Молчанов А.Ю. **Системное программное обеспечение** Спб.: Питер, 2009. – 624 с.
2. Компьютерные сети. Учеб.пособие// Максимов Н. В., Попов И. И. 4-е издание, испр. И доп. – М.: Форум, 2010. 464 с.
3. Компьютерные сети. Учеб.пособие// Кузин А. В., Демин В. М. – 2-е издание – М.: Форум, 2010. 192 с.
4. Сетевые операционные системы / В. Г. Олифер, Н. А. Олифер. – Спб.: Питер, 2009. – 544 с.
5. Бесплатная, виртуальная, электронная, Интернет библиотека www.xserver.ru (дата обращения 13.08.18).
6. Журнал Компьютер Пресс: Тестирование. Безопасность: www.compress.ru (дата обращения 13.08.18).
7. Компьютерная и техническая документация: www.emanual.ru (дата обращения 13.08.18).
8. Образовательный портал: <http://www.edu.sety.ru> (дата обращения 13.08.18).

Перечень материалов, оборудования и информационных источников, используемых в аттестации

- ПК
- ЛВС
- СПО Assembler
- СПО Borland C++ Builder

Спецификация практического задания

Назначение

Спецификацией устанавливаются требования к содержанию и оформлению вариантов оценочного средства.

Практическое задание входит в состав комплекса оценочных средств и предназначено для *текущего контроля* и оценки знаний и умений обучающихся, соответствующих основным показателям оценки знаний и умений обучающихся, соответствующих основным показателям оценки результатов подготовки по программе 09.02.07 Информационные системы и программирование рабочей программы *МДК 01.04 Системное программирование*

Условия аттестации: контроль проводится после изучения тем 1-5 учебной дисциплины в форме лабораторной работы.

1. Время контроля:

выполнение 1 час

2. Структура варианта практического задания

Основная задача: оценка знаний и умений обучающихся, соответствующих основным показателям оценки результатов подготовки по программе учебной дисциплины.

Краткая характеристика

Для реализации личностного потенциала обучающимся предлагаются задания разных уровней.

Задание первого варианта состоит из 1 задачи, выполнение которой является пошаговым действием в достижении результата. Задача данного варианта носит репродуктивный характер.

Задание второго варианта состоит из 1 задачи, в которой указаны исходные данные для выполнения действия и средства выполнения, но отсутствует комментарий к выполнению. Задача данного варианта носит частично-поисковый характер.

Третий вариант включает 1 задачу, в которой указаны исходные данные. Обучающийся должен выбрать средства выполнения, описать алгоритм действий для достижения результата. Задания носят поисковый характер.

Задание

Вариант 1

1. Изучить программную модель микроконтроллера, используя раздел «Теоретическое введение».
2. Изучить инструкцию пользователя симулятора микроконтроллера, используя соответствующий подраздел «Теоретического введения».
3. Изучить команды пересылки. Изучение каждой команды проводить следующим образом:
 - Открыть окно «Дисассемблер».
 - В пункте меню «Правка» выбрать опцию «Вставка команды».
 - В открывшемся окне «Вставить инструкцию» ввести мнемокод соответствующей команды пересылки, описание которой представлено в меню «Справка», и нажать на кнопку ОК. Записать в отчет строку о введенной в окно «Дисассемблер» команде.
 - Открыть окна «Процессор», «Регистры», «Дамп памяти данных». Просмотреть содержимое этих окон.
 - Выполнить команду пересылки, выбрав в меню «Запуск команды» опцию «Шаг».
 - Записать в отчет изменения в окнах «Процессор», «Регистры», «Дамп памяти данных».
 - Для повторного выполнения команды в меню «Запуск команды» выбрать опцию «Сброс».
4. Написать и исследовать работу программы пересылки байта из ячейки памяти программ с адресом 71H в регистр R1 банка 1.
 3. Функциональная модель микроконтроллера.
 4. Содержимое регистров и ячеек памяти окон «Процессор», «Регистры», «Дамп памяти данных» до и после выполнения каждой команды пересылки в соответствии с таблицей 1.3.
 5. Содержимое регистров и ячеек памяти окон «Процессор», «Регистры», «Дамп памяти данных» и памяти программ до и после выполнения каждой команды разработанной программы в соответствии с таблицей 1.4.

Таблица 1 - Результаты выполнения команд

| № | Команда | Код | Выполняемая операция | Содержимое используемых регистров и ячеек памяти до и после выполнения | | Пояснение |
|-----|----------|-----|---|--|-------------------------|-----------|
| | | | | До | После | |
| 1 | MOV A,R0 | E8 | Пересылка байта данных из регистра R0 в аккумулятор A | A/00 PC/00 PSW/00 | A/F2 PC/01 PSW/01 | |
| 2 | ... | ... | ... | ... | ... | ... |
| ... | ... | ... | ... | ... | ... | ... |
| ... | ... | ... | ... | ... | ... | ... |

Вариант 2

1. Исследовать команды записи в стек и считывания из стека: PUSH P1; PUSH 01; PUSH PSW; POP PSW; POP 01; POP P1.
2. Исследовать команды входа и выхода из подпрограмм.
3. Написать и исследовать работу подпрограммы возведения в квадрат числа в аккумуляторе.
4. Написать и исследовать программу опроса содержимого двух младших разрядов порта P1 и перехода в зависимости от их состояния к одной из четырех подпрограмм, начальные адреса которых находятся в ячейках 21H, 25H, 3AH и 2FH.

1. Загрузить последовательность команд: MOV P1, #3FH; MOV R1, #C3H; MOV A, R1; PUSH P1; PUSH 01; PUSH PSW; SETB RS1; MOV P1, #1H; MOV 01H, #02H; POP PSW; POP 01; POP P1.

2. Выполнить последовательность команд в пошаговом режиме.

3. Записать в отчет в соответствии с таблицей 2.2 выполняемые команды и содержимое изменяемых регистров и ячеек памяти.

4. Загрузить, начиная с адреса 0000, последовательность команд: ACALL 05; LJMP 20; NOP; NOP; NOP; RET.

5. Выполнить программу, состоящую из данной последовательности команд в пошаговом режиме.

6. Записать в отчет в соответствии с таблицей 2.2 выполняемые команды и содержимое изменяемых регистров и ячеек памяти.

7. Разработать и ввести подпрограмму возведения в квадрат числа с использованием команды MUL AB. Для обращения к подпрограмме использовать команду ACALL.

8. Убедиться в работоспособности подпрограммы.

9. Записать в отчет в соответствии с таблицей 2.2 выполняемые команды и содержимое изменяемых регистров. Определить время выполнения подпрограммы.

10. Разработать программу опроса содержимого двух младших разрядов порта P1 и перехода в зависимости от их состояния к одной из четырех подпрограмм, начальные адреса которых находятся в ячейках 21H, 25H, 3AH и 2FH

11. Задавая в порт различные кодовые комбинации в используемых разрядах, проверить работоспособность программы.

Таблица 2 - Результаты выполнения команд

| № | Команда | Код | Выполняемая операция | Содержимое регистров и памяти до и после выполнения | | Пояснение |
|---|----------|-----|---|---|--------------------------|-----------|
| | | | | До | После | |
| 1 | MOV A,R0 | E8 | Пересылка байта данных из регистра R0 в аккумулятор A | A/00 R0/F2 0000/F2 | A/F2 R0/F2 0000/F2 | |
| 2 | ... | ... | ... | ... | ... | ... |

Вариант 3

1. Разработать и ввести в окно «Дисассемблер» программу сложения двух произвольно выбранных однобайтовых чисел.
2. Осуществляя запуск программы в режиме «Шаг», убедиться в правильности ее выполнения, используя окна «Дисассемблер». «Процессор», «Регистры». Результаты наблюдений свести в табл.3.3. Определить время выполнения программы.
3. Разработать и ввести в окно «Дисассемблер» программу умножения двух произвольно выбранных однобайтовых чисел.
4. Осуществляя запуск программы в режиме «Шаг», убедиться в правильности ее выполнения, используя окна «Дисассемблер». «Процессор», «Регистры». Обратит внимание на формируемые флаги. Результаты наблюдений свести в таблице 3.3. Определить время выполнения программы.

Таблица 3 - Результаты выполнения команд

| № | Команда | Код | Выполняемая операция | Содержимое регистров и памяти до и после выполнения | | Пояснение |
|---|----------|-----|---|---|--------------------------|-----------|
| | | | | До | После | |
| 1 | MOV A,R0 | E8 | Пересылка байта данных из регистра R0 в аккумулятор A | A/00 R0/F2 0000/F2 | A/F2 R0/F2 0000/F2 | |
| 2 | ... | ... | ... | ... | ... | ... |

5. Разработать и ввести в окно «Дисассемблер» программу деления произвольно выбранных однобайтовых чисел.
6. Осуществляя запуск программы в режиме «Шаг», убедиться в правильности ее выполнения, используя окна «Дисассемблер». «Процессор», «Регистры». Результаты наблюдений свести в таблицу. Определить время выполнения программы.
7. Исследовать команды сдвига вправо и влево. Результаты наблюдений свести в таблицу.
8. Загрузить из каталога *Examples* программу «Временная задержка.txt».
9. Используя меню *Запуск*, установить точку останова программы на команде с адресом 0007.
10. Запустить программу в непрерывном режиме (опция *Запуск*).

11. В окне «Процессор» считать информацию о времени выполнения и числе циклов исполняемой программы.
12. Используя длительности исполнения команд программы, рассчитать время исполнения программы.
13. Ввести изменения в исполняемой программе для получения временной задержки, заданной преподавателем.
14. Запустить модернизированную программу и определить время ее выполнения в соответствии с п.п. 8 – 10. Сравнить полученные данные с расчетным значением.

Система оценки знаний

Отлично: выполнен вариант 3. Ошибки отсутствуют (допущена незначительная ошибка).

Хорошо: выполнен вариант 3, допущено не более 3-х неточностей/ошибок; выполнен вариант 2, ошибки отсутствуют (допущены незначительные недочеты/ошибки, но не более 3-х).

Удовлетворительно: выполнен вариант 3, допущено 4-5 неточностей/ошибок; выполнен вариант 2, допущено не более 4 неточностей/ошибок; выполнен вариант 1, ошибки отсутствуют (допущено не более 3-х неточностей/ошибок).

Неудовлетворительно: допущено большее количество ошибок; задания выполнены частично или не выполнены.

Трудоемкость

Вариант (для всех вариантов)

| Трудоемкость выполнения/решения, мин (час) | Количество задач, вопросов |
|---|----------------------------|
| | 1 |
| Одной задачи | 60мин |
| Всего задания | 60 мин |

КОМПЛЕКТ ЭКЗАМЕНАЦИОННОГО ЗАДАНИЯ

Экзаменационный билет № _1_

1. Поясните понятие системного программирования.
2. Написать и исследовать работу программы пересылки байта из ячейки памяти программ с адресом 71H в регистр R1 банка 1 с использованием регистра DPTR и в регистр R2 банка 3 с использованием регистра PC. (Assembler).
3. Проверить является ли строка палиндромом. (Палиндром - это выражение, которое читается одинаково слева направо и справа налево). (Borland C++ Builder).

Экзаменационный билет № _2_

1. Приведите принципы функционирования систем программирования.
2. Исследовать команды записи в стек и считывания из стека: PUSH P1; PUSH 01; PUSH PSW; POP PSW; POP 01; POP P1 (Assembler).
3. Преобразовать строку таким образом, чтобы в ее начале были записаны слова, содержащие только цифры, потом слова, содержащие только буквы, а затем слова, которые содержат и буквы и цифры. (Borland C++ Builder).

Экзаменационный билет № 3

1. Поясните назначение и функции компилятора.
2. Написать и исследовать работу подпрограммы возведения в квадрат числа в аккумуляторе (Assembler).
3. Преобразовать строку таким образом, чтобы буквы каждого слова в ней были отсортированы по возрастанию. (Borland C++ Builder).

Экзаменационный билет № 4

1. Дайте определение понятиям загрузчика, отладчика, приведите функции загрузчика.
2. Написать и исследовать программу опроса содержимого двух младших разрядов порта P1 и перехода в зависимости от их состояния к одной из четырех подпрограмм, начальные адреса которых находятся в ячейках 21H, 25H, 3AH и 2FH. (Assembler).
3. Преобразовать строку таким образом, чтобы цифры каждого слова в ней были отсортированы по убыванию. (Borland C++ Builder).

Экзаменационный билет № _5_

1. Изобразите блок-схему разработки программ в архитектуре «клиент-сервер».
2. Разработать и исследовать программы сложения операндов (Assembler).
3. Произвести пословный перевод всех слов строки. Подстановочный словарь может содержать не более 10 слов, можно не учитывать изменение форм слова. (Borland C++ Builder).

Экзаменационный билет № _6_

1. Поясните механизм подготовки и выполнения программ на языке Ассемблер.
2. Разработать и исследовать программу умножения операндов (Assembler).
3. Структурированный тип: Фирма

Поля: название фирмы, Ф.И.О. директора, количество сотрудников, адрес, уставной капитал.

Функции:

од количества фирм с клавиатуры

олнение массива структур с клавиатуры

Печать массива структур на экране

Вычисление среднего уставного капитала

Поиск структуры с заданными Ф.И.О. директора. (Borland C++ Builder).

Экзаменационный билет № _7_

1. Перечислите и охарактеризуйте режимы работы микропроцессора.
2. Разработать и исследовать программу деления операндов (Assembler).
3. **Структурированный тип:** абонент телефонной связи.

Поля: Ф.И.О. абонента, номер телефона, тип соединения (индивидуальный, спаренный, коллективного пользования), адрес владельца

Функции:

Подсчет количества абонентов с клавиатуры

Заполнение массива структур с клавиатуры

Печать массива структур на экране

Поиск абонента с заданным номером

Подсчет количества абонентов с телефоном индивидуального пользования. (Borland C++ Builder).

Экзаменационный билет № _8_

1. Перечислите типы операторов языка ассемблер.
2. Исследовать и модернизировать программу формирования временной задержки (Assembler).

3. Структурированный тип: Перечень комплектующих деталей компьютера

Поля: тип детали (например, видеокарта, процессор и т. п), марка, производитель, цена, срок гарантийного обслуживания

Функции:

Заполнение массива структур с клавиатуры

Печать массива структур на экране

Вычисление средней стоимости одного компьютера

Поиск комплектующей детали с минимальным гарантийным сроком обслуживания

Вывод перечня комплектующих заданного типа. (Borland C++ Builder).

Экзаменационный билет № 9

1. Как происходит использование моделей памяти и сегментации при создании программ.
2. Исследовать программу возведения в квадрат числа (Assembler).
3. **Структурированный тип: Средство связи**
Поля: тип, радиус действия, цена одной минуты связи.

Функции:

- Заполнение массива структур с клавиатуры
- Печать массива структур на экране
- Вычисление средней стоимости одного часа связи для средств заданного типа
- Поиск средства связи с максимальным радиусом действия. (Borland C++ Builder).

Экзаменационный билет № 10

1. Перечислите и охарактеризуйте типовые ошибки программирования.
 2. Исследовать программу перевода двоичного числа в двоично-десятичное число (Assembler).
 3. **Структурированный тип: товар**
Поля: Название, фирма-производитель, входная цена одной единицы товара, количество, торговая надбавка в процентах
- Функции:
- Заполнение массива структур с клавиатуры
 - Печать массива структур на экране
 - Поиск товара с минимальной ценой заданного производителя
 - Вычисление общего количества товаров. (Borland C++ Builder).

Экзаменационный билет № 11

1. Запишите и поясните общий формат машинной команды.
2. Разработать и исследовать подпрограмму вычисления синуса угла в диапазоне 0 - 80° с шагом 10°. (Assembler).
3. Структурированный тип: читатель
Поля: Ф.И.О., номер билета, количество книг на руках, дата перерегистрации(число, месяц, год)

Функции:

Заполнение массива структур с клавиатуры

Печать массива структур на экране

Поиск читателя с наиболее поздней датой перерегистрации

Вычисление общего количества книг на руках. (Borland C++ Builder).

Экзаменационный билет № 12

1. Приведите структуру и особенности работы обработчиков прерываний.
2. Разработать программу десятичной коррекции (Assembler)
3. Структурированный тип: книга
Поля: автор, название, год издания, число страниц

Функции:

Заполнение массива структур с клавиатуры

Печать массива структур на экране

Подсчет книг заданного автора

Поиск книги с наиболее поздним годом издания. (Borland C++ Builder).

Экзаменационный билет № _13_

1. Как происходит организация взаимодействия резидентных программ.
2. Разработать программу перевода двоичного числа в двоично-десятичное, используя рекуррентное соотношение
 $d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0 = d_0 + 2(d_1 + 2(d_2 + 2(d_3 + 2(d_4 + 2(d_5 + 2(d_6 + 2 d_7))))))$ (Assembler).

3. Структурированный тип: тарифный план сотовой связи

Поля: Название фирмы-провайдера, название тарифа, абонентская плата, количество бесплатных минут, входящих в абонентскую плату, стоимость минуты разговора в дневное и вечернее время

Функции:

Заполнение массива структур с клавиатуры. Печать массива структур на экране

Поиск тарифа с максимальной абонентской платой. Подсчет оптимального по цене тарифа с условием, что абонент планирует наговаривать в месяц не более некоторого числа минут, заданного с клавиатуры. (Borland C++ Builder).

Экзаменационный билет № _14_

1. Приведите порядок операций по защите программ от копирования и несанкционированного доступа.
2. Разработать программу перевода двоично-десятичного числа в двоичное (Assembler).
3. Написать программу просмотра и редактирования простейшей базы данных о студентах. Базу данных считывать из файла. Приложение должно состоять из двух диалогов. В первом диалоге редактируется число записей в базе, имя и фамилия одного (текущего) студента. Второй диалог предназначен для просмотра всей базы в виде таблицы. Высота таблицы должна автоматически вычисляться по текущему разрешению экрана. (Borland C++ Builder).

Экзаменационный билет № 15

1. Как осуществляются арифметические операции над данными в ASCII - и BCD – форматах.
2. Разработать программу временной задержки для датчика движения (Assembler).
3. Написать программу простейшего графического редактора, позволяющего рисовать (добавлять, удалять, редактировать свойства) фигуры из разработанной системы классов в окне созданного приложения. Изучить структуру приложения Windows, использующего OpenGL. Изучить архитектуру, синтаксис команд, примитивы OpenGL. Написать программу рисования простых трехмерных тел. (Borland C++ Builder).

Экзаменационный билет № 16

1. Приведите примеры мнемонической записи команд языка Ассамблер.
2. Разработать программу вычисления времени до завершения процесса (Assembler).
3. Описать класс точки с характеристиками (x, y) – вещественные координаты точки и методами ввода, вывода и нахождения расстояния от точки до начала координат. (Borland C++ Builder).

Экзаменационный билет № _17_

1. Приведите порядок операций отладки программных модулей с использованием специализированных программных средств.
2. Разработать программу управления семисегментным индикатором (Assembler).
3. Описать класс точки, добавив к нему методы возврата координат X и Y. С помощью этого класса решить следующие прикладные задачи: даны точки $A(x_1, y_1)$ и $B(x_2, y_2)$. Определить, лежат ли длина отрезков OA и OB (O – начало координат) в заданном интервале $[m_1, m_2]$. (Borland C++ Builder).

Экзаменационный билет № _18_

1. Поясните механизм взаимодействия Turbo Assembler и Borland Pascal.
2. Разработать программу определения координат в пространстве (Assembler).
3. Описать класс прямоугольника с характеристиками a и b – длины сторон, и методами нахождения периметра, площади и диагонали прямоугольника. Решить следующие задачи с использованием описанного класса: имеется прямоугольный участок со сторонами A и B. Определить стоимость работ по установке забора вокруг участка, если установки забора длиной 10 метров требует расходов в N рублей. (Borland C++ Builder).

Экзаменационный билет № 19

1. Поясните основные правила синтаксиса языка C++.
2. Разработать программу определения расстояния до объекта (Assembler).
3. Описать класс параболы at^2+bt+c с характеристиками a , b и c , и методами определения координаты t_0 – вершины параболы и значения функции $y = at^2+bt+c$ в произвольной точке t . Для определения значения в вершине параболы воспользоваться тем, что значение производной в этой точке равняется нулю, то есть $2at+b=0$. Решить следующие задачи: спрос на продукцию предприятия снижается по левой ветви параболы t^2-4t+9 . Определить, в какой момент времени t_0 спрос будет минимальным и чему он будет равен (в тыс. ед.) (Borland C++ Builder).

Экзаменационный билет № 20

1. Приведите общие правила построения программ для Windows (на примере C++).
2. Разработать программу управления автомобильным тахометром (Assembler).
3. Имеются данные об успеваемости 10-ти студентов (средние баллы). Найти средний балл по группе. (Borland C++ Builder).

Экзаменационный билет № 21

1. Приведите пример простейшего приложения, реализующего обработку сообщений.
2. Имеется команда:
ECS: 100 B8 04 30 05 00 30 CB
Что делает данная программа?
(Assembler).
3. Формируются несколько групп переключателей по 2 – 3 переключателя и статическое окно. В каждый момент времени только один из переключателей в группе может быть выбран. Информация об изменении состояния каждого переключателя должна отображаться в статическом окне. (Borland C++ Builder).

Экзаменационный билет № 22

1. Опишите функции обработки сообщений от драйвера «мыши», для C++ Builder.
2. Напишите программу, выполняющую следующую операцию:
 - Пересылка значения 25H в регистре AL
 - Сдвиг содержимого регистра AL на 1 бит влево
 - Пересылка значения 15H в регистр BL
 - Умножение содержимого регистра AL на содержимое регистра BL(Assembler).
3. Формируются несколько радиокнопок и статическое окно. В каждый момент времени только одна из радиокнопок должна быть нажата. Информация об изменении состояния каждой кнопки должна отображаться в статическом окне. (Borland C++ Builder).

Экзаменационный билет № 23

1. Приведите механизм отладки программных модулей с использованием специализированных программных средств.
2. Для сегментов кода, данных и стека даны имена CDSEA, DATSEA и STKSEA соответственно. Сформируйте директиву ASSUME. (Assembler).
3. Формируются однострочный текстовый редактор, простой список выбора и стандартная кнопка. Нажатие на кнопку должно помещать введенную пользователем в редакторе строку в список выбора. (Borland C++ Builder).

Экзаменационный билет № 24

1. Приведите порядок операций для оптимизации программных модулей.
2. Напишите программу для вычисления 12 чисел Фибоначчи:
1,1,2,3,5,8,13,...(каждое число в последовательности представляет собой сумму двух предыдущих чисел). Для организации цикла используйте команду LOOP. (Assembler).
3. Формируются список выбора с окном редактирования и статическое окно. В статическом окне должна отображаться информация о выбираемом пользователем элементе списка. (Borland C++ Builder).

Экзаменационный билет № 25

1. Приведите порядок операций для тестирования программных модулей.
2. Напишите программу для вычисления суммы первых 15 нечетных чисел Фибоначчи с первыми 5 четными числами Фибоначчи. Суммы должны вычисляться в подпрограммах. (Assembler).
3. Формируются три стандартные кнопки и статическое окно. Две кнопки управляют размерами третьей: нажатие на них пропорционально увеличивает и уменьшает ее размеры. Информация об изменении размеров кнопки должна отображаться в статическом окне. (Borland C++ Builder).

Экзаменационный билет № 26

1. Таймеры в ОС Windows, связанные с ними сообщения, функции API Windows обработки этих событий.
2. Напишите программу вывода на экран набора символов ASCII – кода (Assembler).
3. Формируются несколько стандартных кнопок и статическое окно. Информация о нажатии каждой кнопки должна отображаться в статическом окне. (Borland C++ Builder).

Экзаменационный билет № 27

1. Опишите функции обработки клавиатурных сообщений в ОС Windows, функции API Windows, для C++.
2. Напишите программу, запрашивающую ввод имени, а затем отображающую в середине экрана введенное имя (Assembler).
3. Имеются данные об объемах продаж магазина за 12 месяцев. Найти суммарный и среднемесячный объемы продаж. (Borland C++ Builder).

Экзаменационный билет № 28

1. Поясните механизм взаимодействия Turbo Assembler и Borland C++.
2. Напишите команды для BIOS INT 13H, выполняющие чтение одного сектора в область памяти INDISK, с дисковода A, головки 0, дорожки 6 и сектора 3 (Assembler).
3. Имеются данные о численности 10 населенных пунктов района. Найти количество населенных пунктов, количество жителей в которых ниже среднего по району. (Borland C++ Builder).

Экзаменационный билет № 29

1. Приведите пример программирования операций над файлами, каталогами и дисками.
2. Напишите команды для BIOS INT 13H, выполняющие запись трех секторов из области памяти OUTDISK на дисковод B, головку 0, дорожку 8 и сектор 1 (Assembler).
3. Имеются данные о прибыли предприятия за последние пять лет. Найти, в какие годы прибыль была максимальной и минимальной. (Borland C++ Builder).

Экзаменационный билет № 30

1. Расшифруйте понятия загрузчики и редакторы связей.
2. Программа MAINPRO должна вызвать подпрограмму SUBPRO. В программе MAINPRO определены переменные QTY как DB, VALUE как DW и PRICE как DW. Подпрограмма SUBPRO должна разделить VALUE на QTY и записать частное а PRICE. Постройте работающую программу и проверьте ее (Assembler).
3. Имеются данные о температуре воздуха за неделю (по дням), определить перепад температур за неделю. (Borland C++ Builder).

Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основная

1. Белева, Л. Ф. Программирование на языке С++ [Электронный ресурс] : учебное пособие / Л. Ф. Белева. — Электрон. текстовые данные. — Саратов : Ай Пи Эр Медиа, 2018. — 81 с. — 978-5-4486-0253-5. — Режим доступа: <http://www.iprbookshop.ru/72466.html>
2. Костюкова, Н. И. Программирование на языке Си [Электронный ресурс] : методические рекомендации и задачи по программированию / Н. И. Костюкова. — Электрон. текстовые данные. — Новосибирск : Сибирское университетское издательство, 2017. — 160 с. — 978-5-379-02016-3. — Режим доступа: <http://www.iprbookshop.ru/65289.html>
3. Устинов, В. В. Основы алгоритмизации и программирование. Часть 2 [Электронный ресурс] : конспект лекций / В. В. Устинов. — Электрон. текстовые данные. — Новосибирск : Новосибирский государственный технический университет, 2013. — 32 с. — 978-5-7782-2337-0. — Режим доступа: <http://www.iprbookshop.ru/44675.html>
4. Фёдорова Г.Н. Разработка программных модулей программного обеспечения компьютерных систем: учебник.-М.:ИЦ Академия, 2016 - 13 экз.
5. Учебники по программированию <http://programm.ws/index.php>
6. С++ для начинающих, <http://mycpp.ru/cpp/book/>

Дополнительная

1. Семакин И.Г., Шестаков А.П. Основы алгоритмизации и программирования/ И.Г. Семакин ОИЦ «Академия» – М.: Издательский центр «Академия», 2016
2. Семакин И.Г., Шестаков А.П. Основы алгоритмизации и программирования. Практикум/ И.Г. Семакин ОИЦ «Академия» – М.: Издательский центр «Академия», 2016
3. Голицына, О.Л., Основы алгоритмизации и программирования: Учебное пособие / О.Л. Голицына, И.И. Попов – М.: Форум: Инфра-М, 2011.
4. Канцедал, С.А. Алгоритмизация и программирование : Учебное пособие / С.А. Канцедал.. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2015. - 352 с.
Микрюков, В.Ю. Алгоритмизация и программирование: Учебное пособие / В.Ю. Микрюков. - Рн/Д: Феникс, 2015. - 304 с.
5. Незнанов, А.А. Программирование и алгоритмизация: Учебник для студентов учреждений высшего профессионального образования / А.А. Незнанов; Науч. ред. В.П. Кутепов. - М.: ИЦ Академия, 2014. - 304 с.

4. Новичков, В.С. Алгоритмизация и программирование на Турбо Паскале / В.С. Новичков, Н.И. Парфилова. - М.: ГЛТ, 2015. - 438 с.
5. Эпштейн М.С. Практикум по программированию: учебное пособие для сред. проф. образования. – М.: Издательский центр «Академия», 2009.
6. Лесневский А.С. Объектно-ориентированное программирование для начинающих. – М.: Бином. Лаборатория знаний, 2009.
7. Ахо А., Хопкрофт Дж., Ульман Дж., Построение и анализ вычислительных алгоритмов. – М.: Мир, 1989. – 369с.
8. Никлаус Вирт. Алгоритмы и структуры данных. – Санкт-Петербург: «Невский диалект», 2001.
9. Альсведе Р., Вегенер И. Задачи поиска.– М.: Мир, 1982. – 368 с.
10. Бауэр Ф.Л., Гооз Г., Информатика. Вводный курс, в 2-ух ч. – М., Мир, 1981. – 368с.
11. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. – М.: Мир, 1982. – 416 с.
12. Дал У., Дейкстра Э., Хоор К. Структурное программирование. – М.: Мир, 1975.
13. Калинин А.Г., Мацкевич И.В. Универсальные языки программирования. Семантический подход.– Радио и связь, 1991.
14. Кристофидес Н. Теория графов. Алгоритмический подход.–М.: Мир, 1978. – 432 с.
15. Лисков Б., Гатэг Дж. Использование абстракций и спецификаций при разработке программ. – М.: Мир, 1989.
16. Лэнгсам Й., Огенстайн М., Тененбаум А. Структуры данных для персональных ЭВМ.– М.: Мир, 1989. – 588с.

Интернет-источники:

1. [Электронный ресурс] <http://www.codenet.ru>
2. [Электронный ресурс] <http://www.chemisk.narod.ru/html/algorithm01.html>
3. Университетская библиотека ONLINE: <http://biblioclub.ru>
4. Научная электронная библиотека eLIBRARY.RU
5. Университетская информационная система РОССИЯ: <http://uisrussia.msu.ru/>.