**АРХАНГЕЛЬСКИЙ КОЛЛЕДЖ ТЕЛЕКОММУНИКАЦИЙ (ФИЛИАЛ) СПбГУТ**

**(АКТ (ф) СПбГУТ)**

**ОТЧЕТЫ ПО ЛАБОРАТОРНЫМ РАБОТАМ**

**ПО СИСТЕМНОМУ ПРОГРАММИРОВАНИЮ**

Студент: Маоатканов А.А.

Группа: ИСПП-21

Преподаватель: Садовский Р.В.

Архангельск 2025

# Лабораторная работа №1 Изучение процесса разработки линейных алгоритмов на языке ассемблера

1. Цель работы
   1. Изучить процесс разработки линейных алгоритмов на языке ассемблера;
   2. Научиться выполнять вычисление математических выражений на языке ассемблера;
   3. Закрепить навык отладки приложений в MS Visual Studio
2. Ответы на контрольные вопросы
   1. Для чего применяется команда mov?

Ответ: Для копирования данных из источника (переменная С++) в приёмник (ассемблерная вставка).

* 1. Какие арифметические команды применяются в языке ассемблера и какое у них назначение?

Ответ: ADD – сложение, SUB – вычитание, INC – инкремент, DEC – декремент, MUL – умножение, DIV – деление.

* 1. Какие побитовые команды применяются для ускорения умножения и деления и когда они применимы?

Ответ: << и >> - сдвиги влево и вправо соответственно. Применять можно с двоичными данными.

* 1. Каков размер в байтах регистров общего назначения EAX, AX, AH, AL?

Ответ: EAX – 32 бит, AX – 16 бит, AH – 8 бит, AL – 8 бит.

* 1. Какие способы обнуления регистров могут применяться в языке ассемблера?

Ответ: XOR, SUB, MOV.

* 1. Можно ли записать значение переменной типа int в регистр AX и почему?

Ответ: В зависимости от размера регистра.

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки линейных алгоритмов на языке ассемблера;
   2. В ходе проделанной лабораторной работы научился выполнять вычисление математических выражений на языке ассемблера;
   3. В ходе проделанной лабораторной работы закрепил навык отладки приложений в MS Visual Studio.

# Лабораторная работа №2 Изучение процесса разработки разветвляющихся алгоритмов на языке ассемблера

1. Цель работы
   1. Изучить процесс разработки разветвляющихся алгоритмов на языке ассемблера;
   2. Закрепить навык отладки приложений в MS Visual Studio.
2. Ответы на контрольные вопросы
   1. Какие логические команды поддерживаются в языке asm?

Ответ: AND, OR, XOR, NOT, TEST.

* 1. Какая команда осуществляет сравнение двух операндов?

Ответ: CMP.

* 1. Какие операторы условного перехода имеются в языке asm?

Ответ: JE/JZ, JNE/JNZ, JA/JNBE, JAE/JNB и т.д.

* 1. Какой оператор безусловного перехода имеется в языке asm?

Ответ: JMP.

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки разветвляющихся алгоритмов на языке ассемблера;
   2. В ходе проделанной лабораторной работы закрепил навык отладки приложений в MS Visual Studio.

# Лабораторная работа №3 Изучение процесса разработки циклов со счётчиком на языке ассемблера

1. Цель работы
   1. Изучить процесс разработки циклов со счётчиком на языке ассемблера;
   2. Закрепить навык отладки приложений в MS Visual Studio.
2. Ответы на контрольные вопросы
   1. Какие команды asm требуются для организации цикла со счётчиком?

Ответ: LOOP.

* 1. Как на языке ассемблера реализовать досрочный выход их цикла?

Ответ: JE, JZ, JNE, JNZ.

* 1. Какие команды asm требуются, чтобы увеличить значение счётчика на 1 и на большее значение (для увеличения значения счётчика)?

Ответ: INC.

* 1. Какие команды asm требуются, чтобы уменьшить значение счётчика на 1 и на большее значение (для уменьшения значения счётчика)?

Ответ: DEC.

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки циклов со счётчиком на языке ассемблера;
   2. В ходе проделанной лабораторной работы закрепил навык отладки приложений в MS Visual Studio.

# Лабораторная работа №4 Изучение процесса разработки разветвляющихся алгоритмов на языке ассемблера

1. Цель работы
   1. Изучить процесс разработки циклов с предусловием и постусловием на языке ассемблера;
   2. Закрепить навык отладки приложений в MS Visual Studio.
2. Ответы на контрольные вопросы
   1. Какое минимальное количество раз выполняется цикл с предусловием?

Ответ: Минимальное количество итераций цикла – 0.

* 1. Какое минимальное количество раз выполняется цикл с постусловием?

Ответ: Минимальное количество итераций цикла – 1.

* 1. Какова общая форма цикла с предусловием на языке ассемблера?

Ответ:

метка\_начала\_цикла:  
 ; Проверка условия  
 CMP регистр, значение ; Сравниваем значение в регистре с некоторым значением  
 JE метка\_конца\_цикла ; Если равны, переходим к концу цикла  
 JL метка\_конца\_цикла ; Если меньше, переходим к концу цикла  
 JG метка\_конца\_цикла ; Если больше, переходим к концу цикла  
  
 ; Тело цикла  
 ; ... инструкции, которые нужно выполнить ...  
  
 ; Возврат в начало цикла  
 JMP метка\_начала\_цикла  
  
метка\_конца\_цикла:  
 ; Код, следующий за циклом  
 ; ...

* 1. Какова общая форма цикла с постусловием на языке ассемблера?

Ответ:

начало\_цикла:  
 ; Блок кода, который выполняется в цикле  
 ...  
  
 ; Проверка условия выхода из цикла  
 cmp регистр, значение ; Сравнение значения в регистре с некоторым значением  
 jne начало\_цикла ; Если не равно, переход к началу цикла  
 ; Если равно, выход из цикла

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки циклов с предусловием и постусловием на языке ассемблера;
   2. В ходе проделанной лабораторной работы закрепил навык отладки приложений в MS Visual Studio.

# Лабораторная работа №5 Изучение принципов работы математического сопроцессора

1. Цель работы
   1. Изучить принципы работы сопроцессора и методы его программирования средствами ассемблера.
   2. Закрепить навык отладки приложений в MS Visual Studio.
2. Ответы на контрольные вопросы
   1. Какая команда используется для инициализации сопроцессора?

Ответ: FINIT.

* 1. Какие команды сопроцессора используются для загрузки констант?

Ответ: FLDZ, FLD1, FLDPI, FLDL2T, FLDL2E, FLDLG2, FLDLN2.

* 1. Что такое ST(0)?

Ответ: ST(0) обозначает верхний элемент стека.

* 1. Какие арифметические команды имеются в сопроцессоре?

Ответ: Сложение, вычитание, умножение, деление, а также операции над порядком и мантиссой чисел.

* 1. Какие команды сопроцессора используются для передачи данных?

Ответ: MOV.

1. Вывод
   1. В ходе проделанной лабораторной работы изучил принципы работы сопроцессора и методы его программирования средствами ассемблера.
   2. В ходе проделанной лабораторной работы закрепил навык отладки приложений в MS Visual Studio.

# Лабораторная работа №6 Изучение принципов работы цепочечных команд

1. Цель работы
   1. Изучить принципы обработки цепочек данных средствами ассемблера.
   2. Закрепить навык отладки приложений в MS Visual Studio.
2. Ответы на контрольные вопросы
   1. Для чего в операциях с цепочками используется флаг DF?

Ответ: Флаг направления (DF - Direction Flag) в операциях с цепочками определяет направление обработки данных.

* 1. Для чего используются префиксы в операциях с цепочками данных?

Ответ: В операциях с цепочками данных префиксы используются для нескольких целей, включая разделение, идентификацию, структурирование и улучшение читаемости.

* 1. Какие команды используются для сравнения цепочек данных?

Ответ: Используются команды, такие как CMPS и SCAS.

* 1. Операнды каких размеров могут использоваться в операциях с цепочками данных?

Ответ: В операциях с цепочками данных в ассемблере, операнды могут быть размером в байт, слово (16 бит) или двойное слово (32 бита).

* 1. Какие регистры предназначены для задания источника и приемника в операциях с цепочками данных?

Ответ: ESI, EDI.

1. Вывод
   1. В ходе проделанной лабораторной работы изучил принципы обработки цепочек данных средствами ассемблера.
   2. В ходе проделанной лабораторной работы закрепил навык отладки приложений в MS Visual Studio.

# Лабораторная работа №7 Изучение процесса разработки модулей на языке ассемблера

1. Цель работы
   1. Изучить процесс разработки модулей с использованием ассемблера MASM.
2. Ответы на контрольные вопросы
   1. Из каких этапов состоит процесс разработки программ на языке ассемблера?

Ответ: Cоставление алгоритма, написание исходного кода, трансляция в машинный код, линковка и отладка.

* 1. Из каких сегментов состоит программа на языке ассемблера и как объявляется каждый из сегментов?

Ответ: Сегмент кода, сегмент данных, сегмент стека – .CODE, .DATA, .STACK соответственно.

* 1. Как в программе на языке ассемблера объявить целое число, строку, массив?

Ответ: dw, db, array db 10, 20, 30

* 1. Для чего предназначен сегмент кода программы на языке ассемблера?

Ответ: Сегмент кода в языке ассемблера предназначен для хранения инструкций машинного языка, составляющих исполняемый код программы.

* 1. Что такое MASM?

Ответ: MASM, или Microsoft Macro Assembler, это ассемблер для процессоров семейства x86. MASM считается одним из мощных ассемблеров благодаря поддержке макросов и широкому контролю над оборудованием.

* 1. Каков синтаксис вызова процедур в программах на языке ассемблера?

Ответ: call MyProcedure - Вызов процедуры.

* 1. Какие способы передачи параметров между процедурами используются в программах на языке ассемблера?

Ответ: Через регистры, стек и общую область памяти.

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки модулей с использованием ассемблера MASM.

# Лабораторная работа №8 Дизассемблирование приложений

1. Цель работы
   1. Научиться применять дизассемблеры для изучения и модификации ПО.
2. Ответы на контрольные вопросы
   1. Что такое «дизассемблирование»?

Ответ: Дизассемблирование - это процесс преобразования машинного кода (двоичного файла) в эквивалентный ему код на языке ассемблера.

* 1. Для чего выполняется дизассемблирование программного кода?

Ответ: Дизассемблирование программного кода выполняется для преобразования машинного кода (двоичного представления программы) в ассемблерный язык, который является более понятным для человека.

* 1. Что такое «дизассемблер»?

Ответ: Дизассемблер - это компьютерная программа, которая преобразует машинный код в язык ассемблера.

* 1. Какие существуют программы-дизассемблеры?

Ответ: IDA, Ghidra, Radare 2, Cutter, Sourcer

* 1. Как открыть окно дизассемблированного программного кода в Visual Studio?

Ответ: перейти в меню Отладка (Debug) > Окна (Windows) > Дизассемблирование (Disassembly), или нажать комбинацию клавиш Ctrl + Alt + D.

* 1. Для чего выполняется обфускация кода?

Ответ: Обфускация кода выполняется для защиты программного обеспечения от несанкционированного доступа и анализа.

1. Вывод
   1. В ходе проделанной лабораторной работы научился применять дизассемблеры для изучения и модификации ПО.

# Лабораторная работа №9 Создание проекта в эмуляторе Arduino

1. Цель работы
   1. Научиться создавать имитационные модели проектов, работающих под управлением микроконтроллеров для Arduino;
   2. Получить навыки работы с платой Arduino Uno и макетной платой
2. Ответы на контрольные вопросы
   1. Что такое и для чего используется «Arduino Uno»?

Ответ: Arduino Uno - это плата микроконтроллера, используемая для создания электронных устройств и проектов.

* 1. Что такое «макетная плата» и для чего она используется?

Ответ: Макетная плата или breadboard, – это универсальная плата для сборки и моделирования электронных схем без пайки.

* 1. Для чего предназначен каждый из типов рельс на макетной плате?

Ответ: Силовые рельсы предназначены для подачи питания на схему, а контактные ряды - для соединения компонентов схемы между собой.

* 1. Как соединить элементы на макетной плате последовательно?

Ответ: Чтобы соединить элементы на макетной плате последовательно, нужно подключить выход одного элемента к входу другого, создавая цепочку, по которой ток будет протекать в одном направлении.

* 1. Как соединить элементы на макетной плате параллельно?

Ответ: Чтобы соединить элементы на макетной плате параллельно, нужно подключить их выводы к одним и тем же рядам (вертикальным линиям) на макетной плате.

* 1. В каком порядке должно выполняться подключение элементов на схеме?

Ответ:  В общем случае, элементы подключаются в соответствии с их функциональным назначением и логической структурой схемы, начиная с источника питания и заканчивая выходными элементами.

1. Вывод
   1. В ходе проделанной лабораторной работы научился создавать имитационные модели проектов, работающих под управлением микроконтроллеров для Arduino;
   2. В ходе проделанной лабораторной работы получил навыки работы с платой Arduino Uno и макетной платой.

# Лабораторная работа №10 Разработка скетчей для Arduino

1. Цель работы
   1. Научиться создавать скетчи (программное обеспечение на языке Си) для микроконтроллеров Arduino;
   2. Получить навыки работы с платой Arduino Uno и макетной платой
2. Ответы на контрольные вопросы
   1. Для чего предназначены и когда вызываются функции setup() и loop()?

Ответ: Функция setup() предназначена для инициализации настроек программы Arduino, функция loop() содержит основной код программы, который выполняется циклически, многократно.

* 1. Для чего используется функция pinMode(...), какие параметры она принимает?

Ответ: Функция pinMode() в Arduino используется для настройки режима работы цифрового вывода.

* 1. Для чего используется функция digitalWrite(...), какие параметры она принимает?

Ответ: Функция digitalWrite() в Arduino используется для установки логического уровня (HIGH или LOW) на цифровом выводе (пине) платы.

* 1. Какие функции используются для реализации программной задержки, какие параметры принимают эти функции?

Ответ: Для реализации программной задержки на Arduino Uno используется функция delay(). Принимает время в миллисекундах.

* 1. Что такое «скетч»?

Ответ: Cкетч - это название программного кода, который пишется для управления платой Arduino.

* 1. Как подключить библиотеки к скетчу?

Ответ: Для подключения в самом скетче используется директива #include <имя\_библиотеки.h>.

1. Вывод
   1. В ходе проделанной лабораторной работы научился создавать скетчи (программное обеспечение на языке Си) для микроконтроллеров Arduino;
   2. В ходе проделанной лабораторной работы получил навыки работы с платой Arduino Uno и макетной платой.

# Лабораторная работа №11 Разработка приложений для обработки файлов

1. Цель работы
   1. Научиться использовать файловые потоки в приложении на C#;
   2. Научиться применять классы для работы с файлами в приложениях на C#.
2. Ответы на контрольные вопросы
   1. Какое пространство имен содержит себе классы для работы с файловой системой и файловыми потоками?

Ответ: System.IO

* 1. Для чего предназначены классы StreamReader и StreamWriter?

Ответ: Классы StreamReader и StreamWriter предназначены для работы с текстовыми файлами в .NET.

* 1. Для чего предназначены классы BinaryReader и BinaryWriter?

Ответ: Классы BinaryReader и BinaryWriter предназначены для чтения и записи данных в двоичном формате.

* 1. Какие классы предоставляют информацию о файлах?

Ответ: File и FileInfo.

1. Вывод
   1. В ходе проделанной лабораторной работы научился использовать файловые потоки в приложении на C#;
   2. В ходе проделанной лабораторной работы научился применять классы для работы с файлами в приложениях на C#.

# Лабораторная работа №12 Разработка приложений для поиска файлов

1. Цель работы
   1. Научиться применять классы для работы с файлами и каталогами в приложениях на C#.
2. Ответы на контрольные вопросы
   1. В чем отличие между классами Directory и DirectoryInfo?

Ответ: Класс Directory предоставляет статические методы для работы с каталогами, в то время как DirectoryInfo является классом экземпляра, который предоставляет информацию о конкретном каталоге и его методах.

* 1. В чем отличие между классами File и FileInfo?

Ответ: Класс File предоставляет статические методы для взаимодействия с файлами, в то время как FileInfo предоставляет методы и свойства для работы с конкретным файлом, как с объектом.

* 1. Как получить список файлов и папок определенного каталога?

Ответ: Через Directory. Чтобы получить список файлов, применяется метод GetFiles(), а для списка папок - GetDirectories().

* 1. Как задать шаблон поиска и опции поиска файлов и папок?

Ответ: SearchOption.

* 1. Какие свойства класса FileInfo позволяют получить информацию о файле?

Ответ: Exists, Length, Name, FullName, DirectoryName, Extension

1. Вывод
   1. В ходе проделанной лабораторной работы научился применять классы для работы с файлами и каталогами в приложениях на C#.

# Лабораторная работа №13 Разработка приложений для сортировки файлов

1. Цель работы
   1. Научиться применять классы для работы с файлами и каталогами в приложениях на C#.
2. Ответы на контрольные вопросы
   1. Как проверить существование файла по его имени?

Ответ: File.Exists(Путь к файлу).

* 1. Как проверить существование каталога по его имени?

Ответ: Directory.Exists(Path).

* 1. Какие методы позволяют создавать, удалять, копировать и переносить каталоги?

Ответ: Directory.CreateDirectory(Path), Directory.Delete(Path), Directory.Move(начальный путь, конечный путь).

* 1. Какие методы позволяют создавать, удалять, копировать и переносить файлы?

Ответ: Методы класса File. Create, Delete, Copy, Move.

1. Вывод
   1. В ходе проделанной лабораторной работы научился применять классы для работы с файлами и каталогами в приложениях на C#.

# Лабораторная работа №14 Разработка утилиты для поиска дубликатов файлов

1. Цель работы
   1. Научиться получать и анализировать информацию о файлах в приложениях на C#.
2. Ответы на контрольные вопросы
   1. Какое свойство FileInfo возвращает имя файла?

Ответ: Name.

* 1. Какое свойство FileInfo возвращает расширение файла?

Ответ: Extension.

* 1. Какое свойство FileInfo возвращает полное имя файла?

Ответ: FullName.

* 1. Какое свойство FileInfo возвращает дату изменения файла?

Ответ: LastWriteTime .

1. Вывод
   1. В ходе проделанной лабораторной работы научился получать и анализировать информацию о файлах в приложениях на C#.

# Лабораторная работа №15 Разработка утилиты для просмотра изображений

1. Цель работы
   1. Научиться отображать файлы-изображения в приложениях на C#.
2. Ответы на контрольные вопросы
   1. Для чего предназначен класс BitmapImage?

Ответ: Предназначен для работы с растровыми изображениями.

* 1. Для чего предназначен элемент управления Image?

Ответ: Элемент управления Image в C# предназначен для отображения изображений на форме или странице.

* 1. Для чего предназначен элемент управления ScrollView?

Ответ: Элемент управления ScrollView предназначен для обеспечения прокрутки содержимого, которое не помещается в области отображения.

* 1. Как получить высоту и ширину изображения BitmapImage?

Ответ: PixelWidth, PixelHeight.

1. Вывод
   1. В ходе проделанной лабораторной работы научился отображать файлы-изображения в приложениях на C#.

# Лабораторная работа №16 Разработка утилиты Диспетчер задач

1. Цель работы
   1. Научиться получать и отображать информацию о запущенных процессах в приложениях на C#.
2. Ответы на контрольные вопросы
   1. Какое пространство имен требуется подключить для работы с процессами?

Ответ: System.Diagnostics.

* 1. Какой метод запускает процесс?

Ответ: Process.Start().

* 1. Какой метод завершает процесс?

Ответ: Process.Kill().

* 1. Как получить список запущенных процессов?

Ответ: Process.GetProcesses().

* 1. Как получить процесс по его идентификатору?

Ответ: Process.GetProcessById(processId).

1. Вывод
   1. В ходе проделанной лабораторной работы научился получать и отображать информацию о запущенных процессах в приложениях на C#.

# Лабораторная работа №17 Разработка утилиты Файловый менеджер

1. Цель работы
   1. Научиться применять элементы управления для отображения файлов и папок в приложениях на C#.
2. Ответы на контрольные вопросы
   1. Как получить список дисков?

Ответ: DriveInfo.GetDrives().

* 1. Как получить список файлов?

Ответ: Directory.GetFiles(path, searchPattern).

* 1. Для чего предназначен элемент управления TreeView?

Ответ: Элемент управления TreeView в C# предназначен для отображения иерархических данных в виде дерева.

1. Вывод
   1. В ходе проделанной лабораторной работы научился применять элементы управления для отображения файлов и папок в приложениях на C#.

# Лабораторная работа №18 Разработка утилиты для анализа дисков

1. Цель работы
   1. Научиться получать и отображать статистическую информацию о дисковом пространстве в приложениях на C#.
2. Ответы на контрольные вопросы
   1. Какой класс предоставляет информацию о дисках?

Ответ: DriveInfo.

* 1. Какой класс предоставляет информацию о файлах?

Ответ: FileInfo.

* 1. Какой метод позволяет просуммировать значения в списке?

Ответ: Sum().

* 1. Какие методы позволяют отсортировать список по возрастанию и убыванию?

Ответ: OrderBy и OrderByDescending.

1. Вывод
   1. В ходе проделанной лабораторной работы научился получать и отображать статистическую информацию о дисковом пространстве в приложениях на C#.

# Лабораторная работа №19 Разработка утилиты Архиватор

1. Цель работы
   1. Научиться применять классы для работы с архивами в приложениях на C#.
2. Ответы на контрольные вопросы
   1. Какие пространства имен нужно подключить для работы с архивами?

Ответ: System.IO и System.IO.Compression.

* 1. Какие ссылки нужно подключить для работы с архивами?

Ответ: System.IO.Compression.

* 1. Для чего предназначен класс ZipArchive?

Ответ: Класс ZipArchive в C# предназначен для работы с ZIP-архивами, позволяя создавать, извлекать и управлять файлами внутри архива.

* 1. Для чего предназначен класс ZipFile?

Ответ: Класс ZipFile в C# предназначен для работы с ZIP-архивами. Он предоставляет статические методы для создания архивов из каталогов и извлечения файлов из архивов в каталоги.

* 1. Для чего предназначен класс ZipArchiveEntry?

Ответ: Класс ZipArchiveEntry в C# представляет отдельную запись (файл или директорию) в ZIP-архиве.

1. Вывод
   1. В ходе проделанной лабораторной работы научился применять классы для работы с архивами в приложениях на C#.

# Лабораторная работа №20 Разработка утилиты скринсейвер

1. Цель работы
   1. Научиться разрабатывать оконные приложения-заставки, используя таймер.
2. Ответы на контрольные вопросы
   1. Что такое «утилита»?

Ответ: Это вспомогательная компьютерная программа, предназначенная для выполнения специализированных задач по обслуживанию операционной системы или оборудования, таких как настройка, управление, оптимизация и диагностика.

* 1. Что такое «скринсейвер»?

Ответ: Это компьютерная программа, которая активируется, когда компьютер не используется в течение определенного времени. Основная цель скринсейвера - предотвратить выгорание экрана и продлить срок его службы.

* 1. Какое расширение должно быть у файлов-заставок в Windows?

Ответ: Расширение .src.

1. Вывод
   1. В ходе проделанной лабораторной работы научился разрабатывать оконные приложения-заставки, используя таймер.

# Лабораторная работа №21 Разработка утилиты для вычисления хэш-суммы файлов

1. Цель работы
   1. Научиться вычислять хэш-суммы файлов в приложениях на C#, используя встроенные алгоритмы шифрования.
2. Ответы на контрольные вопросы
   1. Что такое хэш-сумма?

Ответ: Результат работы хэш-функции, которая преобразует данные произвольного размера в строку символов фиксированной длины.

* 1. Для чего применяются хэш-суммы файлов?

Ответ: Хэш-суммы файлов применяются для проверки целостности данных, идентификации файлов и обеспечения их безопасности.

* 1. Какие алгоритмы применяются для вычисления хэш-суммы файлов?

Ответ: MD5, SHA-1, SHA-256, SHA-512.

1. Вывод
   1. В ходе проделанной лабораторной работы научился вычислять хэш-суммы файлов в приложениях на C#, используя встроенные алгоритмы шифрования.

# Лабораторная работа №22 Разработка утилиты Менеджер паролей

1. Цель работы
   1. Научиться выполнять шифрование и дешифрование данных в приложениях на C#, используя встроенные алгоритмы шифрования.
2. Ответы на контрольные вопросы
   1. Что такое «шифрование»?

Ответ: Шифрование - это процесс преобразования данных в такой вид, который делает их нечитаемыми для тех, кто не имеет специального ключа или алгоритма для расшифровки.

* 1. Что такое «дешифрование»?

Ответ: Дешифрование - это процесс преобразования зашифрованной информации (шифротекста) в ее исходный, читаемый вид (открытый текст). Это обратная операция к шифрованию и обычно требует наличия ключа или алгоритма, используемого при шифровании.

* 1. Что такое «AES»?

Ответ: Стандарт симметричного блочного шифрования, используемый для защиты электронной информации.

* 1. Какими могут быть размеры ключа в алгоритме AES?

Ответ: В алгоритме AES допустимы три размера ключа: 128, 192 и 256 бит.

* 1. Какое пространство имен требуется подключить для применения стандартных алгоритмов шифрования?

Ответ: System.Security.Cryptography.

1. Вывод
   1. В ходе проделанной лабораторной работы научился выполнять шифрование и дешифрование данных в приложениях на C#, используя встроенные алгоритмы шифрования.

# Лабораторная работа №23 Изучение процесса разработки DLL

1. Цель работы
   1. Изучить процесс разработки библиотек динамической компоновки на С++.
   2. Научиться применять библиотеки динамической компоновки написанные на других языках программирования в C#.
2. Ответы на контрольные вопросы
   1. Что такое «библиотека динамической компоновки»?

Ответ: Файл, содержащий код и данные, которые могут использоваться несколькими программами одновременно.

* 1. Какое расширение у файлов библиотек?

Ответ: .dll, .lib, .so, и .a.

* 1. Как подключить библиотеку к проекту?

Ответ: Через DllImport.

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки библиотек динамической компоновки на С++.
   2. В ходе проделанной лабораторной работы научился применять библиотеки динамической компоновки написанные на других языках программирования в C#.

# Лабораторная работа №24 Использование потоков

1. Цель работы
   1. Научиться разрабатывать многопоточные приложения на C#;
   2. Научиться создавать и применять фоновые и основные потоки и выполнять обмен данных между ними в программах на C#.
2. Ответы на контрольные вопросы
   1. В чем отличие потока от процесса?

Ответ: Процесс – это более тяжеловесная структура, представляющая собой запущенную программу с собственным адресным пространством, в то время как поток – это более легковесная единица выполнения, работающая в рамках процесса и делающая возможной многозадачность внутри одной программы.

* 1. Какие преимущества дает многопоточная архитектура?

Ответ: Повышение отзывчивости приложений, более эффективное использование ресурсов процессора и улучшенную структуру программы.

* 1. Какие существуют основные средства синхронизации потоков?

Ответ: Оператор lock.

* 1. Каким образом на однопроцессорных компьютерах исполняются многопоточные приложения?

Ответ: Путём временного разделения процессорного времени между разными потоками.

* 1. Для чего в С# используется класс Thread?

Ответ: Используется для создания и управления потоками выполнения, что позволяет реализовать многопоточность в приложении.

1. Вывод
   1. В ходе проделанной лабораторной работы научился разрабатывать многопоточные приложения на C#;
   2. В ходе проделанной лабораторной работы научился создавать и применять фоновые и основные потоки и выполнять обмен данных между ними в программах на C#.

# Лабораторная работа №25 Разработка сетевого приложения с использованием сокетов

1. Цель работы
   1. Научиться реализовывать и запускать асинхронные операции на C#;
   2. Научиться выполнять вычисления, используя асинхронные операции на C#.
2. Ответы на контрольные вопросы
   1. Что такое «сокет»?

Ответ: Конечная точка соединения, используемая для обмена данными между процессами, которые могут находиться как на одном компьютере, так и на разных, соединенных сетью.

* 1. Каков алгоритм работы сервера, использующего сокеты?

Ответ: Создание сокета, привязка сокета, **прослушивание,** принятие соединения, обмен данными, закрытие соединения, обработка ошибок.

* 1. Каков алгоритм работы клиента, использующего сокеты?

Ответ: Создание сокета, установление соединения, обмен данными, закрытие соединения.

* 1. Какие пространства имен требуется подключить для работы с сокетами?

Ответ: System.Net.Sockets и System.Threading.Tasks.

* 1. Какие параметры требуется указать при создании сокета?

Ответ: Семейство адресов, тип сокета и протокол.

* 1. Как выполнить получение данных с использованием сокетов?

Ответ: Метод Receive у объекта Socket.

1. Вывод
   1. В ходе проделанной лабораторной работы научился реализовывать и запускать асинхронные операции на C#;
   2. В ходе проделанной лабораторной работы научился выполнять вычисления, используя асинхронные операции на C#.

# Лабораторная работа №26 Сетевое программирование сокетов

1. Цель работы
   1. Закрепить навыки работы с сетевыми сокетами на C#;
2. Ответы на контрольные вопросы
   1. Чем отличаются технологии UDP и TCP?

Ответ: TCP обеспечивает надежную, упорядоченную и подтвержденную передачу данных, в то время как UDP является более быстрым, но менее надежным протоколом, не гарантирующим ни доставку, ни порядок пакетов.

* 1. Каков алгоритм работы сервера, использующего сокеты?

Ответ: Создание сокета, привязка сокета, **прослушивание,** принятие соединения, обмен данными, закрытие соединения, обработка ошибок.

1. Вывод
   1. В ходе проделанной лабораторной работы закрепил навыки работы с сетевыми сокетами на C#.

# Лабораторная работа №27 Разработка многопоточных приложений

1. Цель работы
   1. Изучить процесс разработки многопоточных приложений на C#;
2. Ответы на контрольные вопросы
   1. Какие методы предоставляет класс Parallel для обработки данных?

Ответ: Parallel.For и Parallel.ForEach.

* 1. Зачем необходимо синхронизировать потоки и использовать потокобезопасные операции?

Ответ:  Для предотвращения проблем, возникающих при одновременном доступе нескольких потоков к общим ресурсам.

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки многопоточных приложений на C#;

# Лабораторная работа №28 Обмен данными

1. Цель работы
   1. Изучить процесс разработки сетевых приложений с использованием SignalR на C#;
2. Ответы на контрольные вопросы
   1. Что такое SignalR?

Ответ: SignalR – это библиотека для ASP.NET, предназначенная для упрощения разработки веб-приложений, использующих веб-события в реальном времени.

* 1. Каким образом осуществляется обмен данными между клиентом и сервером SignalR?

Ответ: Обмен данными между клиентом и сервером SignalR осуществляется через концентраторы (hubs).

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки сетевых приложений с использованием SignalR на C#;

# Лабораторная работа №29 Работа с буфером экрана

1. Цель работы
   1. Изучить процесс разработки консольных приложений с продвинутым пользовательским интерфейсом;
2. Ответы на контрольные вопросы
   1. Какие основные элементы управления используются в Terminal.Gui?

Ответ: Окна (Windows), метки (Labels), кнопки (Buttons), текстовые поля (TextFields), выпадающие списки (ComboBoxes), списки (ListViews), полосы прокрутки (ScrollBars), рамки (Frame), таблицы (Table) и другие.

* 1. Как в Terminal.Gui реализуются диалоговые окна?

Ответ: В MessageBox.Query/MessageBox.ErrorQuery.

* 1. Как настроить цветовую схему консольного приложения в Terminal.Gui?

Ответ:

ColorScheme colorScheme = new ColorScheme()

{

Normal = Application.Driver.MakeAttribute(Color.Green, Color.Black),

Focus = Application.Driver.MakeAttribute(Color.Blue, Color.Black),

};

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки консольных приложений с продвинутым пользовательским интерфейсом;

# Лабораторная работа №30 Разработка приложения для загрузки и отправки данных по сети

1. Цель работы
   1. Изучить процесс разработки приложений с использованием технологии gRPC;
2. Ответы на контрольные вопросы
   1. В чем заключаются преимущества и недостатки gRPC?

Ответ: Основные преимущества включают высокую производительность, четко определенные интерфейсы, поддержку различных языков и автоматическую генерацию кода. Недостатки gRPC включают сложности с браузерными приложениями (требуется gRPC-Web или транскодирование JSON), меньшую популярность по сравнению с REST и необходимость использования Protocol Buffers.

* 1. Для чего используются файлы .proto?

Ответ: Файлы .proto в C# используются для определения структуры данных и сервисов, используемых в Protocol Buffers (protobuf) и gRPC.

1. Вывод
   1. В ходе проделанной лабораторной работы изучил процесс разработки приложений с использованием технологии gRPC;