Министерство общего и профессионального образования Ростовской области

государственное бюджетное профессиональное образовательное учреждение

Ростовской области

«Таганрогский колледж морского приборостроения»

|  |
| --- |
| Методические рекомендации для прохождения учебной практики |
| профессионального модуля ПМ.01  «Разработка программных модулей программного обеспечения для компьютерных систем» |
| для специальности 09.02.07 «Программирование в компьютерных системах» |

2024

Методические рекомендации разработаны в целях организации и проведения учебной практики в рамках профессионального модуля ПМ.01 «Разработка программных модулей программного обеспечения для компьютерных систем». В ходе прохождения учебной практики обучающийся на основании исходного задания на практику :

–  производит выработку требований к программному обеспечению;

**–** производит проектирование программного обеспечения с использованием специализированных программных пакетов;

– разработку основной технологической документации.

По каждой теме практики, предоставляется отчет.

После прохождения практики, сдается итоговый зачет.

**Описание предметной области**

**Описание предметной области**

Предметная область "Разработка программного модуля для учета заявок на ремонт оргтехники" относится к области обслуживания и ремонта технического оборудования в офисных условиях.

Оргтехника, такая как принтеры, копировальные аппараты, сканеры и другое офисное оборудование, требует регулярного обслуживания и ремонта, чтобы оставаться в рабочем состоянии. В офисах часто возникают проблемы с оргтехникой, которые нужно оперативно решать.

Задача разработки такого программного модуля состоит в создании удобного и эффективного инструмента для учета, отслеживания и управления заявками на ремонт оргтехники, что помогает минимизировать время простоя оборудования и обеспечивает бесперебойную работу в офисных условиях.

Программный модуль для учета заявок на ремонт оргтехники предназначен для автоматизации процесса приема и обработки заявок на ремонт от сотрудников офиса или других пользователей. Он позволяет упростить и ускорить процесс решения проблем с оргтехникой.

Основные функции и возможности модуля включают:

1. Заявка на ремонт: это информация, предоставленная клиентом о неисправности оргтехники, которая требует ремонта. Клиент оставляет новую заявку. Заявка может содержать данные о виде оргтехники, модели, описании проблемы, личную информацию (ФИО клиента и номер телефона). Клиент может отредактировать свою заявку.

2. Регистрация заявки: этот процесс включает приём и регистрацию заявки оператором сервисного центра в системе учёта. Важными аспектами регистрации являются присвоение уникального идентификатора заявке, сохранение информации о заявке.

3. Обработка заявки: процесс, включающий анализ заявки, определение её приоритетности и назначение исполнителя (мастера) для задачи. В процессе обработки может потребоваться дополнительная информация или уточнение деталей проблемы у клиента.

4. Исполнение заявки: фактическое выполнение ремонта оборудования. В этом этапе назначенный мастер ремонтирует оргтехнику, вносит необходимые изменения или заменяет неисправные запчасти. Важно отметить, что на этом этапе могут возникать необходимость заказа запчастей или координации работ с другими специалистами.

5. Отчётность и информирование: важной составляющей учёта заявок на ремонт является фиксация и отчёт о выполненной работе. После завершения ремонта мастер должен предоставить отчёт о проделанной работе, включая информацию о затраченных ресурсах (время, запчастях) и оказанной помощи.

6. Мониторинг и анализ: этот этап предполагает контроль и анализ процесса учёта заявок на ремонт. Важно отслеживать и анализировать время обработки заявок, качество выполненных работ, расходы и прочие параметры, которые могут помочь в оптимизации и улучшении процесса.

**Техническое задание**

1. Общие сведения

1.1. Наименование проекта: Разработка программного модуля для учета заявок на ремонт оргтехники.

1.2. Заказчик: ООО "Полюс".

1.3. Исполнитель: Сервисный центр "IT-Сan".

2. Функциональные требования

2.1. Возможность добавления заявок в базу данных с указанием следующих параметров:

- Номер заявки;

- Дата добавления;

- Вид оргтехники;

- Модель;

- Описание проблемы;

- ФИО клиента;

- Номер телефона;

- Статус заявки (новая заявка, в процессе ремонта, завершена).

2.2. Возможность редактирования заявок:

- Изменение этапа выполнения (готова к выдаче, в процессе ремонта, ожидание запчастей);

- Изменение описания проблемы;

- Изменение ответственного за выполнение работ.

2.3. Возможность отслеживания статуса заявки:

- Отображение списка заявок;

- Получение уведомлений о смене статуса заявки;

- Поиск заявки по номеру или по параметрам.

2.4. Возможность назначения ответственных за выполнение работ:

- Добавление мастера к заявке;

- Отслеживание состояния работы и получение уведомлений о ее завершении;

- Мастер может добавлять комментарии на форме заявки и фиксировать информацию о заказанных запчастях и материалах.

2.5. Расчет статистики работы отдела обслуживания:

- Количество выполненных заявок;

- Среднее время выполнения заявки;

- Статистика по типам неисправностей.

3. Нефункциональные требования

3.1. Кроссплатформенность:

- Поддержка работы на ОС семейства Windows.

3.2. Безопасность:

- Логин и пароль для доступа к приложению;

- Доступ к данным должен быть ограничен в зависимости от роли пользователя.

3.3. Удобство использования:

- Простой и интуитивный интерфейс;

- Информативные уведомления и подсказки.

3.4. Производительность:

- Приложение должно иметь быстрый доступ к данным;

- Минимальное время отклика на запросы пользователя.

4. Требования к реализации

4.1. Язык программирования: на усмотрение разработчика

4.2. СУБД: на усмотрение разработчика

5. Требования к документации

5.1. Техническое задание на разработку программного модуля.

5.2. Руководство системному программисту.

*Файлы для загрузки в базу данных находятся в ресурсах с пометкой import*

**Исходное задание на проектирование и разработку**

**Часть 1**

Проанализировать техническое задание, составить краткую спецификацию разрабатываемого модуля выделить входные и выходные данные; сформировать основной алгоритм решения учета заявок на ремонт оргтехники в виде блок-схемы в соответствии с техническим заданием. Детализировать в виде алгоритма одну из функций (расчета количества заявок, среднего времени ремонта).

Алгоритмы представить одним из способов:

* Алгоритм в виде блок-схемы выполнить по правилам, установленным ГОСТ 19.701.
* Алгоритм в виде таблиц выполнить по правилам, установленным ГОСТ 2.105.
* Алгоритм в виде текстового описания выполнить по правилам, установленным ГОСТ 24.301.

Разработать интерфейс программного модуля по составленному алгоритму в среде разработки в соответствии технического задания. Реализовать последовательности алгоритма по этапам (выходные данные должны соответствовать алгоритму, обрабатывающему входные данные). Реализовать алгоритм с использованием всех необходимых данных. В качестве источников данных для реализации алгоритмов используйте динамические списки или массивы в вашем коде, если не реализовывается БД.

Для работы с разными сущностями используйте разные формы, где это уместно.

Все компоненты системы должны иметь единый согласованный внешний вид, соответствующий руководству по стилю, а также следующим требованиям:  
−последовательный пользовательский интерфейс, позволяющий перемещаться между существующими окнами в приложении (в том числе обратно, например, с помощью кнопки «Назад»);

−соответствующий заголовок на каждом окне приложения.

Выполнить исходный код модуля в соответствии гайдлайну:

идентификаторы должны соответствовать соглашению об именовании, например (CodeConvention), стилю CamelCase (для C# и Java), snake\_case (для Python) и <https://its.1c.ru/db/v8std#browse:13:-1:31> (для 1C).

Допустимо использование не более одной команды в строке.  
Необходимо использовать комментарии для пояснения неочевидных фрагментов кода. Запрещено комментирование кода. Хороший код воспринимается как обычный текст. Не используйте комментарии для пояснения очевидных действий. Комментарии должны присутствовать только в местах, которые требуют дополнительного пояснения.

Реализовать программные обработки исключительных ситуаций в приложении. Уведомляйте пользователя о совершаемых им ошибках или о запрещенных в рамках задания действиях, запрашивайте подтверждение перед удалением, предупреждайте о неотвратимых операциях, информируйте об отсутствии результатов поиска и т.п. Окна сообщений соответствующих типов (например, ошибка, предупреждение, информация) должны отображаться с соответствующим заголовком и пиктограммой. Текст сообщения должен быть полезным и информативным, содержать полную информацию о совершенных ошибках пользователя и порядок действий для их исправления. Также можно использовать визуальные подсказки для пользователя при вводе данных.

Выполнить отладку модуля.

Выполнить отладку программного обеспечения с использованием инструментальных средств. Сохранить и представить результаты в скриншотах.

Определить наборы входных данных и выполнить функциональное тестирование модуля по определенному сценарию. Провести тестирование для проверки функциональности программы (хотя бы 1 тест на 1 функцию). Использовать инструментальные средства для тестирования. Представить результаты тестирования в виде протокола тестирования, в соответствии со стандартами.

**Часть 2**

На основе задания демонстрационного экзамена Вам необходимо спроектировать ER-диаграмму для учета заявок на ремонт оргтехники. Обязательна 3 нормальная форма с обеспечением ссылочной целостности. При разработке диаграммы обратите внимание на согласованную осмысленную схему именования, создайте необходимые первичные и внешние ключи, определите ограничения внешних ключей, отражающие характер предметной области.  
ER - диаграмма должна быть представлена в формате удобном для просмотра и содержать таблицы, связи между ними, атрибуты и ключи (типами данных на данном этапе можно пренебречь) проведение анализа поставленной задачи и проектирования базы данных (ERD модели) с применением case-средств;

Создайте все необходимые сущности, определите отношения, создайте ограничения на связи между сущностями (при наличии всех связей), приведите базу данных к 3НФ (при наличии всех сущностей и связей).

Создайте базу данных, используя предпочтительную платформу, на сервере баз данных, которую Вам предоставили. Создайте таблицы основных сущностей, атрибуты, отношения и необходимые ограничения.

Выполните названия таблиц и полей в едином стиле, согласно отраслевой документации.

Заказчик системы предоставил файлы с данными (с пометкой import в ресурсах) для переноса в новую систему. Заполните базу данных.

Создайте запросы к базе данных и сформируйте отчеты с выводом необходимых данных в соответствии с заданием.

Выполните резервное копирование БД, сохраните полученные результаты.

Выберите принцип регистрации пользователей в системе учета заявок на ремонт оборудования в соответствии с функциональными обязанностями.

Создайте группы пользователей. Выполните реализацию уровней доступа для различных категорий пользователей.

**Задание 1**

**Тема** Анализ технического задания. Разработка проектной документации

**Цель** «Исследовать основные принципы разработки программного обеспечения»

**Время выполнения** – 12 ч

##### **Оборудование:**

– оборудование лаборатории;

*–*методические рекомендации к выполнению заданий,справочная литература или доступ в сеть Интернет

**Задание**

Проанализировать техническое задание, составить краткую спецификацию разрабатываемого модуля выделить входные и выходные данные; сформировать основной алгоритм решения учета заявок на ремонт оргтехники в виде блок-схемы в соответствии с техническим заданием. Детализировать в виде алгоритма одну из функций (расчета количества заявок, среднего времени ремонта).

Алгоритмы представить одним из способов:

* Алгоритм в виде блок-схемы выполнить по правилам, установленным ГОСТ 19.701.
* Алгоритм в виде таблиц выполнить по правилам, установленным ГОСТ 2.105.
* Алгоритм в виде текстового описания выполнить по правилам, установленным ГОСТ 24.301.

**Содержание отчета**

1. Тема задания
2. Анализ предметной области задания
3. Схемы, поясняющие процесс проектирования (диаграмма вариантов использования, блок-схемы алгоритмов, диаграмма деятельности и т.д)

**Критерии оценки:**

* **зачет** выставляется студенту: если задание выполнено, отчет оформлен в соответствии с требованиями, студент способен пояснить проделанную работу;
* **незачет** выставляется студенту: если задание не выполнено, отсутствует отчет, студент не может пояснить проделанную работу

**Задание 2**

**Тема** Проектирование базы данных на основе анализа предметной области

**Цель** «Произвести разработку БД информационной системы»

**Время выполнения** – 12 ч

##### **Оборудование:**

– оборудование лаборатории;

*–*методические рекомендации к выполнению заданий,справочная литература или доступ в сеть Интернет

**Задание**

На основе задания демонстрационного экзамена Вам необходимо спроектировать ER-диаграмму для учета заявок на ремонт оргтехники. Обязательна 3 нормальная форма с обеспечением ссылочной целостности. При разработке диаграммы обратите внимание на согласованную осмысленную схему именования, создайте необходимые первичные и внешние ключи, определите ограничения внешних ключей, отражающие характер предметной области.  
ER - диаграмма должна быть представлена в формате удобном для просмотра и содержать таблицы, связи между ними, атрибуты и ключи (типами данных на данном этапе можно пренебречь) проведение анализа поставленной задачи и проектирования базы данных (ERD модели) с применением case-средств;

Создайте все необходимые сущности, определите отношения, создайте ограничения на связи между сущностями (при наличии всех связей), приведите базу данных к 3НФ (при наличии всех сущностей и связей).

Создайте базу данных, используя предпочтительную платформу, на сервере баз данных, которую Вам предоставили. Создайте таблицы основных сущностей, атрибуты, отношения и необходимые ограничения.

Выполните названия таблиц и полей в едином стиле, согласно отраслевой документации.

Заказчик системы предоставил файлы с данными (с пометкой import в ресурсах) для переноса в новую систему. Заполните базу данных.

**Содержание отчета**

1. Выбор программных средств разработки
2. ER - диаграмма должна быть представлена в формате .pdf или .vsdx и содержать таблицы, связи между ними, атрибуты и ключи. Описание таблиц.

**Критерии оценки:**

* **зачет** выставляется студенту: если задание выполнено, отчет оформлен в соответствии с требованиями, студент способен пояснить проделанную работу;
* **незачет** выставляется студенту: если задание не выполнено, отсутствует отчет, студент не может пояснить проделанную работу

**Задание 3**

**Тема** Разработка модулей программного обеспечения в соответствии с техническим заданием

**Цель** «Произвести разработку модулей программного обеспечения»

**Время выполнения** – 36ч

##### **Оборудование:**

– оборудование лаборатории;

*–*методические рекомендации к выполнению заданий,справочная литература или доступ в сеть Интернет

**Задание**

Произвести разработку приложения в соответствии с техническим заданием.

**Содержание отчета**

1 Описание программного модуля в соответствии с ГОСТ 19.402-78.

2 Схемы поясняющие работу

**Критерии оценки:**

* **зачет** выставляется студенту: если задание выполнено, отчет оформлен в соответствии с требованиями, студент способен пояснить проделанную работу;
* **незачет** выставляется студенту: если задание не выполнено, отсутствует отчет, студент не может пояснить проделанную работу

**Задание 4**

**Тема** «Инспектирование компонент программного продукта на предмет соответствия стандартам кодирования»

**Цель** «Проведение оптимизации и рефакторинга программного решения. Разработка эксплуатационной документации»

**Время выполнения** – 6 ч

##### **Оборудование:**

– оборудование лаборатории;

*–*методические рекомендации к выполнению заданий,справочная литература или доступ в сеть Интернет

**Задание**

В рамках данного задания необходимо оптимизацию программного решения.

Вам необходимо разработать руководство пользователя для вашего приложения, которое описывает последовательность действий для выполнения всех функций вашей системы.

При подготовке документации старайтесь использовать живые примеры и скриншоты вашей системы для более наглядного пояснения шагов работы с различным функционалом.

Обратите внимание на оформление документа: оформите титульный лист, используйте автоматическую нумерацию страниц, разделите руководство на подразделы и сформируйте оглавление, используйте ссылки на рисунки, нумерованные и маркированные списки для описания шагов и т. д.

**Содержание отчета**

1 Решения принятые в рамках оптимизации кода.

2 Произвести разработку руководство пользователя

**Критерии оценки:**

* **зачет** выставляется студенту: если задание выполнено, отчет оформлен в соответствии с требованиями, студент способен пояснить проделанную работу;
* **незачет** выставляется студенту: если задание не выполнено, отсутствует отчет, студент не может пояснить проделанную работу

**Задание 5**

Тема «Тестирование»

**Цель** «Проведение тестирования приложения»

**Время выполнения** – 12 ч

##### **Оборудование:**

– оборудование лаборатории;

*–*методические рекомендации к выполнению заданий,справочная литература или доступ в сеть Интернет

**Задание**

В рамках данного задания необходимо произвести разработку набора тестов и провести тестирование на наличие логических ошибок и ошибок проектирования.

**Содержание отчета**

Тема задания

Программа и методики испытаний

Протоколы испытаний

**Критерии оценки:**

* **зачет** выставляется студенту: если задание выполнено, отчет оформлен в соответствии с требованиями, студент способен пояснить проделанную работу;
* **незачет** выставляется студенту: если задание не выполнено, отсутствует отчет, студент не может пояснить проделанную работу

**Задание 6**

Тема «Сопровождение программного обеспечения»

**Цель** «Выполнение работы по модификации отдельных компонент программного обеспечения в соответствии с потребностями»

**Время выполнения** – 18 ч

##### **Оборудование:**

– оборудование лаборатории;

*–*методические рекомендации к выполнению заданий,справочная литература или доступ в сеть Интернет

**Задание**

В рамках определения модификации программного продукта разработайте документ Руководство системному программисту в соответствии со стандартом ЕСПД. Сохраните итоговый документ с руководством системного программиста в формате текстового документа, используя в качестве названия следующий шаблон: Руководство системного программиста XX, где XX – ваш номер по журналу. Из дополнения к техническому заданию предложите варианты модификации программного обеспечения, предложения представьте в текстовом файле. Добавьте нового пользователя в систему. Создайте новую роль Менеджер. Добавьте функционал согласно должностным инструкциям Менеджера, в соответствии с требованиями заказчика. Установите необходимые компоненты, в рамках требований заказчика на модификацию программного обеспечения, в соответствии с дополнением к техническому заданию. Выполните настройку ПО эксплуатации программного обеспечения. Добавьте функционал согласно с требованиями заказчика. Определите качественные характеристики кода такие как: полнота обработки ошибочных данных, наличие тестов для проверки допустимых значений входных данных, наличие средств контроля корректности входных данных, наличие средств восстановления при сбоях оборудования, наличие комментариев, наличие проверки корректности передаваемых данных, наличие описаний основных функций. Представьте результаты в формате текстового документа

**Содержание отчета**

Все практические результаты должны быть переданы путем загрузки файлов на предоставленный репозиторий системы контроля версий.

Практические результаты:

* исходный код приложения (структура с файлами, не архив);
* исполняемые файлы;
* файл скрипта базы данных;
* прочие графические/текстовые файлы.

Результаты работ загружать в рамках выполнения задания модуля.

**Критерии оценки:**

* **зачет** выставляется студенту: если задание выполнено, отчет оформлен в соответствии с требованиями, студент способен пояснить проделанную работу;
* **незачет** выставляется студенту: если задание не выполнено, отсутствует отчет, студент не может пояснить проделанную работу

**Задание 8**

Тема «Низкоуровневое программирование»

**Цель** «Выполнение разработки отдельных компонент в соответствии со спецификацией»

— углубить и закрепить знания по архитектуре МП платформ х86/64 и навыки по их программированию;

— приобрести практические навыки составления, налаживания и выполнения программ, написанных языком ассемблера для программирования

**Время выполнения** – 18 ч

##### **Оборудование:**

– оборудование лаборатории;

*–*методические рекомендации к выполнению заданий,справочная литература или доступ в сеть Интернет

**Задание**

Постановка задачи

Выполнить задание в 2х вариантах: на masm32( предусмотреть режим отладки) и в VStudio (используя ассемблерные ставки) . Вывод — в упрощенное Windows окно (функция MessageBox). Итого — 2 программы.

Согласно номеру студента в группе выбрать вариант задания и написать на ассемблере программу вычисления одного из выражений

Задание

|  |  |  |  |
| --- | --- | --- | --- |
| 1. 2*d/с* – *сd*; | 8. 2*ab* – 8*c*/*b*; | 15. *d*/3*c* – 15*ac*; | 22. 2*b* – *e*/22*b*; |
| 2. 2*a* – *e*/2*c*; | 9. 8/*d* – 9*d*/*c*; | 16. 2*d/*3*с* – 16*сd*; | 23. *ab*/3*a* – 23*b*; |
| 3. *b*/*a* + *c*/*a*; | 10. 3*e*/*b* + 10*c*; | 17. 2*a*/7 – *d*/17*e*; | 24. *d*/4*a* – 24*d*/*c*; |
| 4. *d*/*b* – *d*/4*c*; | 11. 2*d*/3*b* – 11*c*; | 18. 2*ab* – 18*c*/*d*; | 25. *e*/8*b* + 25*ac*; |
| 5. *e*/3*c* + *ac*; | 12. 3*b* – 12*c*/*d*. | 19. 8*d/b* – 19*d*/*c*; | 26. 4*d/*4*a* – 26*сd*; |
| 6. 2*d/с* – *сd*; | 13. *cb*/3*a* – 13*a*; | 20. 3*d*/*b* + 20*e*; | 27. 8*a*/7 – *d*/27*b*; |
| 7. 2*a*/7 – *c*/7*e*; | 14. *e*/4*b* – *d*/14*c*; | 21. 2*d/a* – 21*bd*; | 28. 4*ac* – 28*c*/*b*, |

где: *a, b, c, d, e* – числа.

Постановка задачи

Выполнить два задания:

1 — математическое выражение;

2 — массив.

Выполнить задание в 2х вариантах: на masm32( предусмотреть режим отладки) и в VStudio (используя ассемблерные ставки) . Вывод — в упрощенное Windows окно (функция MessageBox). Итого — 2 программы. .

Вывод — в упрощенное Windows окно

(функция MessageBox). В первом задании результат сохранить в ячейке памяти и вывести результат и адрес переменной в памяти.

Задание 1 — математическое выражение.

Согласно номеру студента в группе выбрать вариант задания и написать на ассемблере программу вычисления одного из выражений:

|  |  |
| --- | --- |
| 1. ac + b/d + f/e; | 15. g/f + e/d — cb/a; |
| 2. f/e — b/d – a/c; | 16. gf/e + dc/b — a; |
| 3. b/d + f/e — ca; | 17. gf + e/d/c + ba; |
| 4. db – hc +a/e+f; | 18. g + fe/d/cb — a; |
| 5. a/c + bd — efg; | 19. gf/e+dc/b + a; |
| 6. a/b/c — d/e + f; | 20. g/f + (ed)/c+b/a; |
| 7. a/d + c/b + efg; | 21. 2d/a – 21bc+f; |
| 8. ef + d/c — ab; | 22. 4b – a/22c + de; |
| 9. a/b — cd/e+f/g; | 23. ab/4e – 16cde; |
| 10. a + b/c/d + efg; | 24. d/ba – cd/e/f; |
| 11. abc — de/fg; | 25. e/8b + acdef; |
| 12. a/b + cd + e/fg; | 26. 4d/a – сd+ bef; |
| 13. ab + cd/ef — g; | 27. 8a/b + cd – (e/f)g; |
| 14. abcd — ef/g; | 28. 4abc – e/f — d, |

где *a, b, c, d, e, f, g* – числа

Задание 2 — массив.

Согласно номеру студента в группе выбрать вариант задания и написать на ассемблере программу вычисления одного из выражений:

1. Заданы натуральные числа *а, а* …*а* . Указать те из них, в которых остаток от деления на *M* будет равняться *L* (0 <= *L* <= *M* – 1)

2. Задан массив *А* с числом элементов больше 7. Написать программу определения количества элементов массива *А,* которые удовлетворяют условию *L ≥ A* ≥ *M*, где *L* = 9 та *M* = 30.

3. Задан массив *А* с числом элементов больше 7. Написать программу формирования массива *В* из последних 13 элементов массива *А,* которые равняются нулю.

4. Задан массив *А* с числом элементов больше 8. Написать программу формирования массива *В* из элементов массива *А,* которые удовлетворяют условие *А ≤ Е* при *Е* = 12.

5. Задан массив *А* с числом элементов больше 8. Написать программу определения суммы и количества элементов массива *А,* которые удовлетворяют условие *А ≤ Е* при *Е* = – 13.

6. Задан массив *А* с числом элементов больше 7. Написать программу определения количества пар элементов, которые удовлетворяют условие *А ≤ В .*

7. Заданы массивы *А* и *B* с числом элементов больше 7. Написать программу формирования массива *С* по такому правилу: если *А – В* ≤ 0, то *С = A .*

8. Задан массив *А* с числом элементов больше 8. Написать программу определения максимального из отрицательных элементов массива *А*.

9. Задан массив *А* с числом элементов больше 9. Структура массива *А* такая: *X*1, *Y*1; *X*2, *Y*2; …. Написать программу определения количества пар, для которых выполняется условие *Х* >= *Y* .

10. Задан массив *А* с числом элементов больше 7. Написать программу формирования массива *В* из элементов массива *А,* в которых биты 0, 2 и 5 имеют нули.

11. Задан массив *А* с числом элементов больше 8. Написать программу определения суммы элементов массива *А,* для которых биты 2 и 10 совпадают.

12. Задан массив *А* с числом элементов больше 8. Структура массива *А* такая: *X*1, *Y*1; *X*2, *Y*2; … . Написать программу определения минимального *X* .

13. Задан массив *А* с числом элементов больше 8. Написать программу определения минимального из положительных элементов массива *А.*

14. Заданы массивы *А* и *В*. Написать программу формирования массива *С* по такому правилу: если *А + В* >0, то *С = В .*

15. Заданы массивы *А* и *В*. Написать программу определения количества пар элементов, которые удовлетворяют условию *А > В .*

16. Задан массив *А* с числом элементов больше 8. Написать программу определения суммы и количества элементов массива *А,* которые удовлетворяют условию *А > Е* при *Е* = – 10.

17. Задан массив *А* с числом элементов больше 8. Написать программу формирования массива *В* из элементов массива *А,* которые удовлетворяют условие *А > Е* при *Е* = 5.

18. Задан массив *А* с числом элементов больше 9. Написать программу формирования массива *В* из первых 8 положительных элементов массива *А*.

19. Задан массив *А* с числом элементов больше 10*.* Написать программу определения количества элементов массива *А,* которые удовлетворяют условию *L* < *A* ≤ *M*, где *L* = 2 та *M* = 10.

20. Заданы массивы *А и В* с числом элементов больше 8. Написать программу формирования массива *С* по такому правилу: если *А > В ,* то *С = А + В* ; иначе *С = А – В .*

**Пример выполнения a\*b/c**

.686

.model flat, stdcall

option casemap:none

include \masm32\include\windows.inc

include \masm32\include\kernel32.inc

include \masm32\include\user32.inc

includelib \masm32\lib\user32.lib

includelib \masm32\lib\kernel32.lib

.data

\_a1 dd 7

\_b1 dd 96

\_c1 dd 19

\_res1 dd ?

\_title db "Практическая занятие №2 Арифметические выражения",0

strbuf dw ?,0

\_text db "Вывод результата a\*b/c через MessageBox:",0ah,"Результат: %d - целая часть",0ah,0ah,0

.code

start:

mov eax,\_a1

mul \_b1 ; edx,eax

div \_c1

mov \_res1,eax

invoke wsprintf, ADDR strbuf, ADDR \_text, \_res1

invoke MessageBox, NULL, addr strbuf, addr \_title, MB\_ICONINFORMATION

invoke ExitProcess, 0

END start

**Пример выполнения a\*b/c, с использование процедуры**

Листинг с отладочным кодом

.686

.model flat, stdcall

option casemap:none

include \masm32\include\windows.inc

include \masm32\include\kernel32.inc

include \masm32\include\user32.inc

includelib \masm32\lib\user32.lib

includelib \masm32\lib\kernel32.lib

include <\masm32\include\debug.inc>

includelib <\masm32\lib\debug.lib>

count PROTO arg\_c:DWORD, arg\_a:DWORD, arg\_c:DWORD

.data

\_a1 dd 7

\_b1 dd 96

\_c1 dd 19

\_res1 dd ?

\_title db "Практическая занятие №2 Арифметические выражения",0

strbuf dw ?,0

\_text db "Вывод результата a\*b/c через MessageBox:",0ah,"Результат: %d - целая часть",0ah,0ah,0

.code

count proc arg\_a:DWORD, arg\_b:DWORD, arg\_c:DWORD

mov eax,arg\_a

mul arg\_b ; edx,eax

PrintDec EAX, "A\*B"

PrintLine

div arg\_c

PrintDec EAX, "A\*B/C"

PrintLine

mov \_res1,eax

ret

count endp

start:

PrintLine

PrintText "Вычисление выражения "

PrintLine

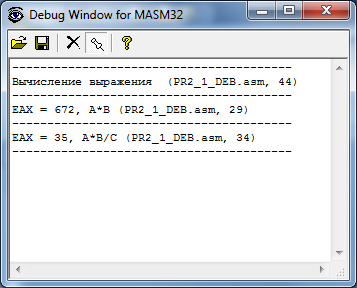
invoke count, \_a1, \_b1, \_c1

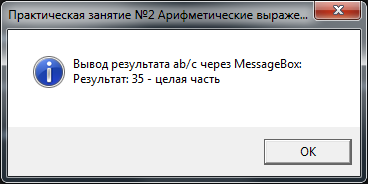
invoke wsprintf, ADDR strbuf, ADDR \_text, \_res1

invoke MessageBox, NULL, addr strbuf, addr \_title, MB\_ICONINFORMATION

invoke ExitProcess, 0

END start





**Листинг без отладочного кода**

.686

.model flat, stdcall

option casemap:none

include \masm32\include\windows.inc

include \masm32\include\kernel32.inc

include \masm32\include\user32.inc

includelib \masm32\lib\user32.lib

includelib \masm32\lib\kernel32.lib

count PROTO arg\_c:DWORD, arg\_a:DWORD, arg\_c:DWORD

.data

\_a1 dd 7

\_b1 dd 96

\_c1 dd 19

\_res1 dd ?

\_title db "Практическая занятие №2 Арифметические выражения",0

strbuf dw ?,0

\_text db "Вывод результата a\*b/c через MessageBox:",0ah,"Результат: %d - целая часть",0ah,0ah,0

.code

count proc arg\_a:DWORD, arg\_b:DWORD, arg\_c:DWORD

mov eax,arg\_a

mul arg\_b ; edx,eax

div arg\_c

mov \_res1,eax

ret

count endp

start:

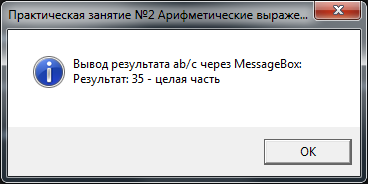
invoke count, \_a1, \_b1, \_c1

invoke wsprintf, ADDR strbuf, ADDR \_text, \_res1

invoke MessageBox, NULL, addr strbuf, addr \_title, MB\_ICONINFORMATION

invoke ExitProcess, 0

END start



Пример выполнения заданий

Задание 2.1 Выражение ab + c/d + e - f

.686

.model flat, stdcall

option casemap:none

include \masm32\include\windows.inc

include \masm32\include\kernel32.inc

include \masm32\include\user32.inc

includelib \masm32\lib\user32.lib

includelib \masm32\lib\kernel32.lib

count PROTO arg\_a:DWORD,arg\_b:DWORD,arg\_c:DWORD,arg\_d:DWORD,arg\_e:DWORD,arg\_f:DWORD

.data

\_a dd 2

\_b dd 3

\_c dd 21

\_d dd 5

\_e dd 10

\_f dd 4

\_res dd ?,0

strbuf dd ?,0

\_text db "Уравнение ab + c/d + e - f",0ah,"Результат: %d",0ah,"Адрес переменной в памяти: %p",0ah,0

\_title db "Практическое занятие процедуры (выражение) (masm32)",0

.code

count proc arg\_a:DWORD,arg\_b:DWORD,arg\_c:DWORD,arg\_d:DWORD,arg\_e:DWORD,arg\_f:DWORD

mov eax,arg\_a

mul arg\_b ; edx,eax

mov esi,eax ; сохранение промежуточного результата

mov eax,arg\_c ;

div arg\_d ;

add eax,esi ;

add eax,arg\_e ;

sub eax,arg\_f ;

mov \_res,eax

ret

count endp

start:

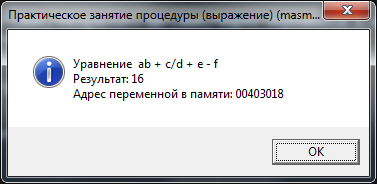
invoke count,\_a,\_b,\_c,\_d,\_e,\_f

invoke wsprintf, ADDR strbuf, ADDR \_text, \_res, ADDR \_res

invoke MessageBox, NULL, addr strbuf, addr \_title, MB\_ICONINFORMATION

invoke ExitProcess, 0

END start



Задание 2.2 Массив с чисел 15-25. Нахождение суммы чисел 2й и 7й бит которых одинаковые

Листинг

.686

.model flat,stdcall

option casemap:none ; отличие строчных и больших букв

include \masm32\include\windows.inc ; файлы структур, констант .

include \masm32\include\user32.inc ; файлы интерфейса .

include \masm32\include\kernel32.inc ; системные функции применений.

includelib \masm32\lib\user32.lib

includelib \masm32\lib\kernel32.lib

.data

\_title db "Практическое занятие массивы. Массивы. masm32", 0

info db "Массив с чисел 15 - 25.",0Ah,0Dh,

"Нахождение суммы чисел, 2-й и 7-й бит которых одинаковые", 0Ah,0Dh,"Сумма чисел : %d",0

x dd 15,16,17,18,19,20,21,22,23,24,25

N equ ($-x)/4 ; N dd 11

res dd 0

buf db 8 dup(?)

.code

start:

mov edx, 0

mov ecx, N ; счетчик числа элементов массива

mov ebx, 0 ; начальный адрес первого элемента в массиве

a0: mov eax, x[ebx] ; запись элемента массива в eax

bt eax, 2 ; выделяем 2-ой бит из eax (значение записывается во флаг CF)

jc a1 ; если CF=1, то перейти на метку a1

jnc a2 ; если CF=0, то перейти на метку a2

a1:bt eax, 7; выделяем 7-ой бит из eax (значение записывается во флаг CF)

jc a3 ; если CF=1, то перейти на метку a3

jnc K ; если CF=0, то перейти на метку K

a2:bt eax, 7; выделяем 7-ой бит из eax (значение записывается во флаг CF)

jnc a3 ; если CF=0, то перейти на метку a3

jc K ; если CF=1, то перейти на метку K

a3: add res, eax ; считаем сумму подходящих элементов массива

K: add ebx, type x ; повторение цикла, пока не закончатся элементы

dec ecx

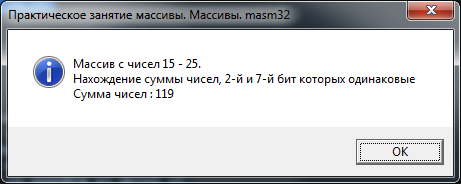
jnz a0

invoke wsprintf,ADDR buf,ADDR info,res;

invoke MessageBox, 0, addr buf, addr \_title, MB\_ICONINFORMATION

invoke ExitProcess, 0

end start



В отчете представить:

— номер и название практического занятия;

— задание;

— текст программы с комментариями к каждой строке программы;

— результат выполнения программы (скриншоты упрощенных окон (MessageBox) и отладчиков);

— особенности выполнения, где описать общий алгоритм (последовательность) выполнения и особенности программы (на что

необходимо с вашей точки зрения уделить внимание).

**Критерии оценки:**

* **зачет** выставляется студенту: если задание выполнено, отчет оформлен в соответствии с требованиями, студент способен пояснить проделанную работу;
* **незачет** выставляется студенту: если задание не выполнено, отсутствует отчет, студент не может пояснить проделанную работу

**Учебно-методическое обеспечение практики**

Основные источники:

1 https://metanit.com/

2 https://www.swrit.ru/doc/espd/19.402-78.pdf