

Самостоятельная работа 6.

Тема. Структура данных - Таблица

Цель:

- получение знаний и умений по использованию типа `struct` языка C++;
- получение навыков по реализации многоэлементных структур данных средствами структуры данных – таблица.

1. Структура данных Таблица. Основные сведения

Таблица – линейная структура данных (массив), элемент которой представляет объект предметной области, который характеризуется несколькими свойствами возможно различного типа. Более простое определение элемента таблицы – это пара типа `<Key, Data>`, где `Key` – ключ, значение которого идентифицирует значение поля `Data` (информационная часть элемента). Ключ используется для доступа к элементам при выполнении операций.

Например, объект Книга, в библиотечной системе может быть определен следующим набором свойств: ISBN, Автор, Название, Издательство, Год издания, Количество страниц, Цена. Массив, элемент которого имеет тип Книга и будет представлять таблицу. Элементы таблицы содержат поле – ключ, значение которого однозначно идентифицирует данные элемента. Так ISBN – ключ для данной книги.

Таблица может быть реализована на статическом и динамическом массивах, на векторе. Структура элемента может быть реализована пользователем на базовом типе `struct` языка C++. Аналогично типу `struct` контейнер `pair` реализует элемент из двух значений, возможно разного типа, и не обязательно простого типа.

STL имеет два контейнера, которые реализуют таблицы: `map` и `multimap`.

Формируемую в практической работе таблицу, можно рассматривать как базу данных, реализованную в оперативной памяти. А разработанные операции – средства управления базой данных.

2. Требования к выполнению практической работы

1. Требуется реализовать три задания. Вариант задачи один для трех заданий и представлен в табл. 11.

В задании 1 таблица формируется на основе статического массива, в задании 2 на основе вектора, в задании 3 на основе контейнера `map` или

multimap. Выбор применяемого контейнера зависит от задачи и определяется разработчиком.

Задания данной практической работы аналогичны заданиям практических работ 1 и 2, алгоритмы операций идентичны, изменяется только структура элемента массива (он имеет сложную структуру), но элементы таблицы одного типа, их можно присваивать. Есть исключение при присваивании таких элементов: если хотя бы одно из полей является указателем.

2. Условие задачи для всех вариантов.

Дано множество из n элементов, структура элемента определен вариантом.

Дан набор задач (операций), которые требуется выполнить над исходным множеством (таблицей). Набор задач определен в варианте.

В определении структуры элемента определено поле, которое является ключом. Несколько элементов могут иметь одинаковые ключи.

Разработать программу, которая управляет таблицей, выполняя операции варианта. Управление выбором операции осуществляется посредством текстового меню.

3. Разработать АТД задачи, по управлению множеством посредством операций, указанных в варианте задания.

В определение АТД включить:

- 1) определение структуры данных: указать тип элемента и тип множества;
- 2) операции:
 - заполнение одного элемента множества;
 - заполнение исходного множества;
 - отображение одного элемента множества;
 - отображение множества;
 - удаление элемента в заданной позиции;
 - добавление нового элемента в таблицу;
 - вставка элемент в заданную позицию;
 - операции, определенные в варианте и выявленные разработчиком как вспомогательные.

2.1. Задание 1

Реализовать АТД управления таблицей, созданной на основе статического массива.

Структура элемента таблицы определена вариантом и представлена в табл. 11.

2.1.1. Требования к выполнению задания 1

1. Создать проект, добавить заголовочный файл определения типа на основе АТД.

Включить в файл заголовка:

- определение типа элемента множества выполнить на основе типа struct;
- определение типа таблицы на базе типа struct, включив в структуру поля: n-текущий размер таблицы, N - максимальный размер таблицы, массив из N элементов;
- определение прототипов операций, определенных в АТД.

2. Требования к реализации операций.

- 1) каждая операция варианта задания должна быть оформлена функцией с параметрами;
- 2) для заполнения отдельного элемента с клавиатуры разработать функцию, которая принимает в качестве параметра пустую запись, а возвращает заполненную;
- 3) наполнение таблицы данными должна выполнять функция вставки или добавления записи, включенные в список операций варианта, функция должна принять через параметр заполненную запись, выполнить действие по вставке или добавлению, согласно алгоритму варианта этой операции;

Примечание. Так как для отладки операций требуется наполнить таблицу данными, а ввод с клавиатуры длительный процесс, предлагается разработать функцию чтения исходных данных из текстового файла (аналогично вводу с клавиатуры). Текстовый файл сформировать в папке проекта, прямо из среды разработки, располагая значение для каждого поля на отдельной строке (так алгоритм чтения будет проще, так как числовые и строковые данные читаются по-разному). Пример формирования текстового файла и код функции, заполнения таблицы представлены на рис. 1.

- 4) алгоритм формирования списка элементов (указано в некоторых вариантах), выполнить на динамическом массиве или векторе.
3. Разработать набор тестов для каждой задачи варианта.
 4. Разработать программу, демонстрирующей выполнение операций над таблицей по разработанным тестам посредством текстового меню.

2.2. Задание 2

Реализовать АТД управления таблицей, используя контейнер `<vector>`, в отчет включить только код и результаты тестирования на векторе.

2.2.1. Требования к выполнению задания

1. Создать новый проект.
2. Скопировать файл заголовка с реализованным АДТ и файл с реализованными операциями. Внести изменения в структуру, определяющую таблицу. Внести изменения в коды операций управления таблицей.
3. Внести изменения в код основной программы, реализованной в задании 1, в связи с изменением структуры элемента и таблицы.
4. Протестировать программу на подготовленных в задании 1 тестах.

```
using namespace std;
struct record {
    int key;
    string data;
};
void readFromFile(string nametf, record* R);
int main()
{
    record R[8];
    readFromFile("A.txt", R);
    for (int i = 0; i < 8; i++)
        cout << R[i].key << ' ' << R[i].data << endl;
    std::cout << "Hello World!\n";
}
void readFromFile(string nametf, record *R) {

    ifstream itf(nametf);
    int i = 0;
    while (!itf.eof()) {
        itf >> R[i].key;
        itf.get();
        getline(itf, R[i].data, '\n');
        i++;
    }
    itf.close();
}
```

Рисунок 1. Код алгоритма чтения данных таблицы из файла

2.3. Задание 3

Реализовать АДТ управления таблицей, используя контейнер `<map>` или `<multimap>`, в отчет включить только код и результаты тестирования на векторе.

2.3.1. Требования к выполнению задания

1. Изучить представление в программе таблицы на основе контейнеров `map` и `multimap` (выбор в соответствии с данными, которые будут представлены в таблице).
2. Ознакомиться с методами управления таблицей `map` или `multimap`.

3. Выбрать вид таблицы, более оптимальной для задачи варианта.
4. Создать новый проект.
5. Скопировать файл заголовка с реализованным АТД и файл с реализованными операциями из задания 1. Внести изменения в структуру, определяющую таблицу. Внести изменения в коды операций управления таблицей.
6. Внести изменения в код основной программы, реализованной в задании 1, в связи с изменением структуры таблицы.
7. Протестировать программу на подготовленных в задании 1 тестах.
8. Оформить отчет.

3. Варианты задач к практической работе 6

Таблица 11. Варианты задач заданий 1, 2, 3 практической работы 6

Примечание. Поле, представляющее ключ, подчеркнуто.

№	Структура элемента таблицы и операции варианта
1.	<p>Студент: <u>номер зачетной книжки</u> (число), фамилия и инициалы, номер группы (текст длиной 10), оценки (успеваемость) (массив из пяти элементов).</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить таблицу с клавиатуры (или из текстового файла); 2) добавить запись в таблицу; 3) сформировать список студентов (номера зачетных книжек), средний балл которых больше 4.0, вставляя значения в порядке убывания баллов (сортировку не применять); 4) удалить из таблицы записи о студентах, средний балл которых меньше заданного значения.
2.	<p>Студент: <u>номер зачетной книжки</u> (число), фамилия и инициалы, номер группы (текст длиной 10), оценки (успеваемость) (массив из пяти элементов).</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить отдельную запись с клавиатуры; 2) вставить новую запись в таблицу сохраняя упорядоченность размещения в порядке возрастания ключа (номер зачетки); 3) найти номер зачетной книжки студента (студентов) с максимальным средним баллом; 4) удалить из таблицы студентов заданной группы.

3.	<p>Экзаменационная ведомость: код института, код дисциплины, фамилия и инициалы преподавателя, количество заполненных в ведомости строк по оценкам сведения по студентам, получившим оценки массив из 30 элементов (фамилия и инициалы, <u>номер зачетной книжки</u>, дата получения оценки, оценка). Оценка 1 выставляется при неявке студента на экзамен, оценка 6 при не допуске к экзамену.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) добавить оценку студенту за экзамен; 2) вставить сведения по студенту за экзамен, учитывая номер зачетки: перед большим номером зачетки; 3) подсчитать количество студентов, не явившихся на экзамен; 4) удалить сведения о студентах, не допущенных на экзамен.
4.	<p>Аэропорт (табло прибытия пассажирских авиарейсов): пункт вылета, номер рейса, дата прилета, время прилета, информация о задержке прилета в часах. Количество строк на табло 10. Рейсов по прилету может быть 20.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) вставить информацию по рейсу в соответствии с датой и временем прибытия; 2) удалить информацию о прибывшем рейсе и сохранить ее в архивной таблице; 3) определить, сколько рейсов было задержано в указанную дату; 4) вывести информацию на табло.

№	Структура элемента таблицы и операции варианта
5.	<p>Аэропорт (управление вылетом пассажирских авиарейсов): пункт назначения, <u>номер рейса</u>, дата вылета, время вылета, время прибытия в пункт назначения, количество свободных мест, информация о задержке вылета в часах. Количество строк на табло отображения 10.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) вставить информацию по новому рейсу в таблицу перед рейсом с большим номером; 2) удалить информацию об вылетевшем рейсе и сохранить ее в архивной таблице; 3) вывести рейсы, готовые к вылету по расписанию (за 2 часа до времени вылета); 4) сформировать список номеров рейсов, вылетающих в заданный пункт назначения.
6.	<p>Аэропорт (расписание авиарейсов): пункт назначения, <u>номер рейса (целое число)</u>, день недели (номер), время вылета/прилета, информация о задержке вылета/прилета в часах.</p> <p><u>Примечание.</u> Номер рейса нечетный – это рейс, вылетающий из аэропорта, четный номер рейса – это рейс, прилетающий в аэропорт.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) добавить расписание по новому рейсу; 2) удалить информацию об отмененном рейсе и сохранить ее в архивной таблице; 3) сформировать список рейсов, вылетающих из аэропорта в заданный день недели и в заданный пункт назначения; 4) подсчитать количество рейсов, прилетающих в аэропорт в заданный день недели.
7.	<p>Учет кадров организации. Сотрудник: <u>табельный номер</u> (трехзначное число), фамилия и инициалы, код занимаемой должности, дата поступления на работу. Создать справочник по должностям – массив из 5 элементов, структура элемента: код (двухзначное число), название должности. Справочник заполните при инициализации.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить с клавиатуры запись по одному сотруднику; 2) вставить в таблицу сведения по сотруднику так, чтобы запись разместилась после последней записи с такой же должностью; 3) заменить у всех сотрудников код заданной должности на новый код (перевели на другую должность); 4) удалить сотрудников, занимающих заданную должность.
8.	<p>Расписание поездов вокзала. Структура элемента расписания: <u>номер поезда</u>, пункт назначения, время отправления. В один пункт назначения может быть несколько поездов в сутки.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить запись по одному элементу расписания с клавиатуры; 2) вставить новую запись так, чтобы записи оказались упорядоченными по возрастанию по времени (сортировку не использовать);

	3) удалить поезд по его номеру; 4) вывести расписание поездов в заданный пункт назначения.
--	---

№	Структура элемента таблицы и операции варианта
9.	<p>Спортивная команда. Структура записи о спортсмене: Фамилия, Имя, Дата рождения, Вес, Рост, Пол, <u>Вид спорта</u>.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить запись по одному спортсмену; 2) вставить запись в конец подсписка таблицы в соответствии с видом спорта; 3) удалить записи о спортсменах, рост которых больше заданного; 4) сформировать список спортсменов команды одного вида спорта.
10.	<p>Учет проданных на поезд билетов. Структура элемента: <u>номер поезда</u>, номер вагона, номер места, дата отправления, стоимость билета. Таблица хранит данные по всем вагонам поезда.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) добавить запись о продаже билета в таблицу; 2) определить количество билетов, проданных на поезд заданного номера и дате отправления; 3) удалить запись из таблицы по проданному билету; 4) вывести сведения о количестве проданных билетов в каждый вагон поезда.
11.	<p>Записная книжка друзей. Структура элемента (записи): Фамилия, Имя, <u>Номер телефона</u>, Дата рождения (три значения), Адрес.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) вставить новую запись в таблицу так, чтобы она стала первой в подписке фамилий записей таблицы, которые начинаются с той же буквы, что и фамилия во вставляемой записи, если нет таких фамилий, то добавить запись в таблицу; 2) удалить запись по заданному номеру телефона; 3) сформировать список друзей, родившихся в заданном месяце (по номеру месяца); 4) найти дату рождения и номер телефона друга по его Фамилии.
12.	<p>Титул учебного плана (УП) направления ВУЗа. Структура записи о плане: шифр направления (например, 090304), название направления, шифр профиля (например, 090304_1), название профиля), год начала действия УП.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить запись по одному учебному плану с клавиатуры; 2) вставить запись в таблицу планов в начало подсписка, с одним и тем же шифром направления; 3) вывести названия профилей и год начала действия учебного плана заданного направления; 4) удалить планы срок действия которых истек при прошествии 4-х лет.
13.	<p>Карточка пользователя библиотеки (выдача книг). Структура записи в карточке: <u>Инвентарный номер</u>, Автор, Название, Дата выдачи, Дата возврата</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить запись по одной взятой книге; 2) удалить записи по заданной дате возврата;

	3) определить, сколько книг не сданы вовремя (по дате возврата); 4) Добавить запись о вновь выданной книге.
--	--

№	Структура элемента таблицы и операции варианта
14.	<p>Учет пациентов поликлиники. Структура записи о пациенте: <u>Номер полиса</u>, фамилия, имя, отчество, дата рождения, номер телефона, дата прививки.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить запись о пациенте поликлиники с клавиатуры; 2) вставить новую запись в таблицу так, чтобы записи были упорядочены по возрастанию номеров полиса (сортировку не использовать); 3) удалить запись по заданному номеру полиса; 4) сформировать список пациентов, не имеющих прививки.
15.	<p>Учет заболеваний пациента. Структура записи о пациенте: <u>Номер полиса</u>, фамилия, имя, отчество, список заболеваний. Структура записи по заболеванию: код заболевания, дата установки диагноза.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить запись по пациенту с клавиатуры, не заполняя список заболеваний; 2) добавить запись о пациенте в таблицу; 3) вставить новую запись по заболеванию в начало списка заболеваний заданного пациента; 4) сформировать список пациентов, которым поставлен диагноз с заданным кодом заболевания.
16.	<p>Учет заболеваний пациента. Структура записи о пациенте: <u>Номер полиса</u>, фамилия, имя, отчество, список заболеваний. Структура записи по заболеванию: код заболевания, дата установки диагноза.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) добавить запись о пациенте в таблицу; 2) вставить новую запись по заболеванию в начало списка заболеваний заданного пациента; 3) сформировать список пациентов, которым поставлен диагноз с заданным кодом заболевания более одного раза; 4) удалить сведения о заболевании у заданного пациента по заданному коду заболевания.
17.	<p>Учет техосмотра автомобилей. Структура записи об автомобиле: <u>Номер</u> (код региона, цифровой код, буквенный код), Модель, Цвет, Сведения о владельце (Фамилия, Имя, Адрес), дата последнего техосмотра.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнение записи по одному автомобилю с клавиатуры; 2) вставить запись по автомобилю в таблицу, так, чтобы сохранялась упорядоченность по возрастанию по дате техосмотра; 3) удалить сведения об автомобиле с заданным номером; 4) сформировать список автомобилей заданной модели и заданного цвета.

№	Структура элемента таблицы и операции варианта
18.	<p>Учет нарушений ПДД. Структура записи о нарушении ПДД: номер автомобиля, фамилия и инициалы владельца, модель, дата нарушения, место нарушения (текстом), статья (КоАП), наказание (сумма штрафа).</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнение записи о нарушении ПДД с клавиатуры; 2) вставить запись в таблицу, упорядочивая записи по не возрастанию по номеру автомобиля; 3) удалить записи по заданному номеру автомобиля; 4) сформировать список сведений по автомобилям, которым назначен штраф по заданной статье.
19.	<p>Справочник банков. Об отдельном банке хранятся данные: наименование, <u>код банка</u>, адрес, форма собственности (коммерческий или государственный), список услуг банка для юридических лиц (<i>справка</i>: предоставлении кредитов, автокредитов, выпуске кредитных, платежных карт, размещении денег на депозитных, текущих счетах, проведении расчетно-кассовых и других операций).</p> <p>Операции:</p> <ol style="list-style-type: none"> 5) заполнение записи по отдельному банку с клавиатуры (список услуг не заполнять); 6) добавить запись по банку в таблицу; 7) вставить в начало списка данные по предоставляемой услуге в список услуг банка; 4) удалить запись о заданном банке; 8) сформировать список государственных банков, предоставляемых ипотечный кредит.
20.	<p>Касса магазина. Структура записи операции по кассе: <u>номер кассы</u>, код товара, количество товара, цена товара, процентная скидка на товар, сумма за товар с учетом скидки.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить запись по одной операции кассы с клавиатуры; 2) вставить записи по отдельной операции, проведенной кассой, в таблицу, располагая ее в начале списка других операций, проведенных этой кассой; 3) вывести список операций, проведенных заданной кассой, с указанием стоимости покупки по каждой операции; 4) удалить записи по операциям, проведенным определенной кассой.
21.	<p>Поисковик туров. Структура записи по туру: <u>страна</u>, дата начала, дата завершения, отель (категория по звездам), тип питания (, «все включено», шведский стол, завтрак в отелях), вид отдыха (пляжный, активный, экскурсионный), стоимость тура.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнение записи по одному туру с клавиатуры; 2) вставить сведения по туру в таблицу, упорядочивая по стране; Новая запись вставляется как последняя запись в подсписок по стране; 3) удалить сведения по завершенным турам;

	4) сформировать список туров в страну по предполагаемой стоимости.
--	--

№	Структура элемента таблицы и операции варианта
22.	<p>Киноафиша города. Структура записи о сеансе: <u>название кинотеатра</u>, название фильма, дата, время начала, стоимость билета.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнение записи по сеансу с клавиатуры; 2) вставить запись по сеансу в таблицу, упорядочивая по названию кинотеатра перед первой записью по этому кинотеатру (сортировку не использовать); Если записи по вставляемому кинотеатру не было, то добавить запись в конец таблицы; 3) удалить записи по заданной дате; 4) вывести список кинотеатров, в которых можно посмотреть заданный фильм (по названию).
23.	<p>Касса кинотеатра. Структура записи о проданном месте: название фильма, дата, время, номер ряда, номер места, номер зала (1-малый, 2-большой). Справочная информация о количестве мест в зале кинотеатра: массив с указанием количества мест всего в каждом зале.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнение записи по одному проданному билету с клавиатуры; 2) вставить запись в таблицу, упорядочивая по номеру зала (сортировку не использовать: новая запись вставляется в начало подписка по залу); 3) удалить записи по заданной дате; 4) определить количество свободных мест на заданный фильм в указанную дату и время.
24.	<p>Расписание занятий группы в ВУЗе. Структура записи по отдельному занятию: <u>номер группы</u>, код дисциплины, день недели, номер пары, номер аудитории. Примечание. Контролировать наложение по аудиториям. Справочник дисциплин оформить в виде массива из записей вида: название дисциплины и его заполнить при инициализации. Кодом дисциплины будет индекс элемента.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнение записи по одному занятию с клавиатуры; 2) вставить запись так, чтобы занятия одного дня недели составляли отдельный подподписок (запись вставлять в начало подподписка); 3) удалить записи из расписания по заданной дисциплине (по коду); 4) вывести расписание по дням недели и с указанием названия дисциплины.
25.	<p>Заказ и доставка товаров из супермаркета. Структура записи о товаре: название, <u>артикул</u>, количество, цена за единицу. Структура записи о заказе: номер заказа, дата заказа, фамилия заказчика, номер телефона, список товаров).</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) определить стоимость покупки по заказу; 2) добавить в список заказа новый товар; 3) добавить новый заказ в список заказов супермаркета; 4) удалить заказ с заданным номером из списка заказов супермаркета.

№	Структура элемента таблицы и операции варианта
26.	<p>Учет занятых и свободных аудиторий в расписании занятий группы ВУЗА.</p> <p>Структура записи по отдельному занятию: <u>номер группы</u>, код дисциплины, день недели, номер пары, номер аудитории.</p> <p>Определить структуру для хранения сведений по аудиториям вуза в соответствии с вашими тестовыми данными.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнение записи по отдельному занятию с клавиатуры; Номер аудитории должен быть выбран из таблицы учета аудиторий; 2) вставить запись так, чтобы занятия одного дня недели составляли отдельный подподписок; Запись вставлять в конец своего подподписки; 3) переместить занятие на другой день недели на свободную пару и в свободную аудиторию; 4) вывести расписание по заданной дисциплины (по коду).
27.	<p>Расчет часовой занятости группы в ВУЗе. Структура записи по отдельному занятию: номер группы, код дисциплины, день недели, номер пары, номер аудитории, код вида занятия. Справочник дисциплин оформить в виде массива из записей вида: название дисциплины и его заполнить при инициализации. Кодом дисциплины будет индекс элемента. Справочник видов занятий и часов, отводимых на занятие в неделю представить массивом вида: (лекция – 2, практическое занятие-2, лабораторная работа - 4, практика - 2). Код вида занятия – это индекс элемента массива.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнение записи по одному занятию с клавиатуры; 2) вставить запись так, чтобы занятия одного дня недели составляли отдельный подподписок; Запись вставлять в начало подподписки; 3) удалить записи из расписания по заданной дисциплине (по коду); 4) рассчитать количество часов в неделю по каждому виду занятия и результат вывести.
28.	<p>Экзаменационная сессия и больно-рейтинговая система. Структура записи о результатах экзамена: код студента (номер зачетной книжки), <u>код дисциплины</u>, балл на экзамене (не более 50), балл за активность (от 0 до 25), оценка за экзамен (значения от 2 до 5).</p> <p>Справочник по дисциплине – это массив, инициализированный названиями дисциплин. Код дисциплины – это индекс элемента массива. Соответствие баллов оценкам: до 16 – 2; от 17 до 45 – 3; от 46 до 60 – 4; от 61 до 75 – 5.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) вставить запись по студенту группы, заполнив поля: код студента, код дисциплины, балл за активность; 2) заполнить поле балл на экзамене заданного студента по заданной дисциплине вводом с клавиатуры; 3) рассчитать оценку: перевести общий балл в оценку от 2 до 5 в соответствии с таблицей; 4) сформировать ведомость по экзамену по заданной дисциплине по группе.

	<u>Примечание.</u> Структура записи ведомости соответствует структуре записи о результатах экзамена.
--	--

№	Структура элемента таблицы и операции варианта
29.	<p>Расчет активности на основе выполнения индивидуальных заданий.</p> <p>Структура записи по выполненным заданиям: <u>код студента</u> (номер зачетной книжки, номер задания, задача 1, задача 2, задача 3).</p> <p>Балл за задачу: задача 1 – 10 баллов; задача 2 -20 баллов; задача 3 -60 баллов.</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнение данных по заданному заданию заданного студента с клавиатуры; 2) вставить запись по студенту в начало таблицы; 3) удалить записи по заданному студенту; 4) сформировать таблицу с результатами выполнения заданий заданным студентом.
30.	<p>Разработка тестовых вопросов. Структура записи вопроса: <u>номер темы</u>, текст вопроса, балл за правильный ответ, список вариантов ответов, номер правильного ответа. По каждой теме вопросов 10 (это условно).</p> <p>Операции:</p> <ol style="list-style-type: none"> 1) заполнить запись по одному вопросу с клавиатуры; 2) вставить запись в таблицу, упорядочивая по номеру темы (сортировку не использовать); 3) сформировать тест, включая в него по два вопроса из каждой темы, выбранных случайно; 4) удалить вопросы по одной теме.

4. Структура отчета

Титульный лист.

Оглавление.

1. Задание 1.

1.1. Условие задачи и требования варианта задания.

1.2. Постановка задачи.

1.3. Модель решения.

Описать модель решения, включив в описание:

1) изобразить рисунком структуру элемента таблицы, представляя ячейки полей и названия полей;

2) изобразить рисунком структуру таблицы, показав все поля;

3) определение АТД;

4) код определений структуры элемента и таблицы.

1.4. Таблицы с подготовленными тестами.

1.5. Код программы.

1.6. Результаты тестирования операций задания (скрины).

2. Задание 2.

2.1. Код программы.

2.2. Результаты тестирования операций задания (скрины).

3. Задание 3.

3.1. Описать формат представления в памяти, используемой таблицы map или multimap.

3.2. Код программы.

3.3. Результаты тестирования операций задания (скрины).

5. Контрольные вопросы

1. Для представления каких данных удобно использовать структуру struct?
2. Какой термин используется для элемента структуры?
3. Какие два вида элементов может содержать определение структуры struct?
4. Какой вид элементов структуры обязательно должен быть описан при ее определении?
5. Какого типа элементы можно включать в структуру?
6. Приведите описание типа, который будет использован в программе для представления календарной даты со свойствами: номер дня, номер месяца, номер года. Номера должны быть представлены целым беззнаковым типом.
7. Календарную дату со свойствами: номер дня, номер месяца, номер года в программе можно определить через строку или структуру. Какое представление даст большую эффективность при доступе к элементу и почему?
8. Какие свойства структур данных общие для массива и struct?
9. Требуется определить тип T для представления следующих данных: ФИО, Дата рождения (должна быть представлена тремя отдельными значениями). Можно ли элемент *Дата рождения* определить непосредственно при определении структуры T?
10. Какой объем памяти будет выделен переменной X?
struct T{int z; char ch; int a}; T X;
Проведите эксперимент, применив функцию sizeof(T).
Объясните различие результатов. Как устранить различие?
11. Приведите оператор вывода для отображения i-ого элемента переменной rec.
struct record{
 unsigned int polis;
 struct {string city, street; unsigned shortint numhouse;} address;
 string phone;

```
};
```

```
vector<record> rec(100);
```

12. Опишите структуру для доступа к тетрадам переменной типа `unsigned int` посредством битовых полей. Приведите код программы, которая выводит шестнадцатеричный код значения переменной.
13. Приведите пример применения типа `union` и `struct` для доступа к отдельным битам переменной.

