

Самостоятельная работа 3.

Тема. Абстрактный тип данных и его реализация на одномерном динамическом массиве и векторе

Цель:

- приобретение умений и навыков определения и реализации абстрактного типа данных задачи в программе;
- приобретение навыков по реализации алгоритмов операций над массивом через аппарат функций языка C++;
- приобретение навыков реализации операций модификации (вставка и удаление элементов) одномерного массива и других операций обработки массива как структуры данных.

1. Требования к выполнению практической работы

Данная работа включает выполнение двух заданий.

1. Реализовать АТД задачи, разработанное в практической работе 1, используя для представления значений множества динамический массив из n элементов.
2. Реализовать АТД задачи, разработанное в практической работе 1, используя для представления значений множества контейнер `<vector>` библиотеки STL языка C++.
3. Оформить отчет.

2. Задание 1

Реализовать АТД задачи, разработанное в практической работе 1, используя для представления значений множества динамический массив.

Выполнить реализацию АТД задачи на динамическом массиве. Для управления динамической памятью использовать функции файла заголовка `stdlib.h`: *malloc*, *free*, *realloc*. Сведения по применению функций представлена в приложении 1.

2.1. Требования к выполнению задания 1

1. В АТД, разработанное в практической работе 1 добавить дополнительные операции варианта, представленного в табл. 7.
2. Создать копию проекта практической работы 1.
3. Внести изменения в структуру, описывающую множество на основе типа `struct` практической работы 1, в связи с применением динамического массива.

4. Внести изменения в операции модификации множества, в связи с применением динамического массива, реализуемого с применением функций *malloc*, *free*, *realloc*.
5. Добавить в АТД операцию: добавление нового элемента в массив значений множества и выполнить ее реализацию с применением функций *malloc*, *free*, *realloc*.
6. Добавить в АТД операцию, определенную в вариантах дополнительного задания этой практической работы в табл. 7 и выполнить ее реализацию.
7. Каждая операция должна быть оформлена отдельной функцией с параметрами.

Примечание. Реализацию АТД можно выполнить в отдельном файле заголовка, разрабатываемого проекта. В основной программе его нужно будет подключить.

8. Разработать набор тестов для тестирования всех операций.
9. Разработать основную программу, демонстрирующую выполнение всех операций над структурой данных на подготовленных тестах. Программа должна управлять всем процессом посредством текстового меню.
10. Выполнить тестирование основной программы на представленных тестах. Представить скрины результатов тестирования.

3. Варианты дополнительных задач

Таблица 7. Варианты дополнительных задач к заданиям практической работы 3

№	Дополнительные операции над элементами структуры
1.	Сформировать новый массив из простых чисел исходного массива.
2.	Сформировать новый массив из тех чисел исходного массива, у которых первая и последняя цифры совпадают.
3.	Сформировать новый массив из чисел исходного массива, цифры в которых упорядочены по убыванию.
4.	Сформировать новый массив из совершенных чисел исходного массива.
5.	Сформировать новый массив из чисел исходного, у которых сумма цифр кратна 7.
6.	Сформировать новый массив из чисел исходного, у которых в записи числа нет цифры 3.

№	Дополнительные операции над элементами структуры
7.	Сформировать новый массив из чисел исходного, у которых цифровой корень однозначное число.
8.	Сформировать новый массив из чисел исходного, у которых старшая цифра является максимальной цифра числа.
9.	Сформировать новый массив из чисел исходного, которые являются палиндромами четной длины.
10.	Сформировать новый массив из чисел исходного, цифры которых упорядочены по возрастанию.
11.	Сформировать новый массив из чисел исходного, цифры которого не образуют последовательность Фибоначчи.
12.	Сформировать новый массив из чисел исходного, двоичный код которых содержит только три единицы.
13.	Сформировать новый массив из чисел исходного, у которых шестнадцатеричный код содержит в старшем разряде цифр А.
14.	Сформировать новый массив из чисел исходного, двоичный код которых содержит два нуля.
15.	Сформировать новый массив из чисел исходного, у которых все цифры одинаковые.
16.	Сформировать новый массив из чисел исходного, в шестнадцатеричный код которого начинается с цифры F.
17.	Сформировать новый массив из чисел исходного, которые являются палиндромами нечетной длины.
18.	Сформировать новый массив из чисел исходного, троичный код которого содержит две двойки.
19.	Сформировать новый массив из чисел исходного, в десятичной записи которых нет цифры 0.
20.	Сформировать новый массив из чисел исходного, произведение цифр которого кратно 7.
21.	Сформировать новый массив из чисел исходного, которые делятся и на младшую и старшую цифры числа.
22.	Сформировать новый массив из чисел исходного, которые делятся на каждую цифру числа.
23.	Сформировать новый массив из чисел исходного, у которых четное количество цифр.
24.	Сформировать новый массив из чисел исходного, которые являются степенью двойки. Требование: для определения степени использовать свой алгоритм, а не логарифм.
25.	Сформировать новый массив из чисел исходного, которые делятся на старшую цифры числа.

№	Дополнительные операции над элементами структуры
26.	Сформировать новый массив из чисел исходного, которые имеют более двух целых положительных делителей.
27.	Сформировать новый массив из чисел исходного, которые в десятичной записи числа содержат цифру 5.
28.	Сформировать новый массив из трехзначных чисел исходного.
29.	Сформировать новый массив из чисел исходного, которые являются простыми числами, у которых старшая и младшая цифры равны.
30.	Сформировать новый массив из чисел исходного, которые являются числами Мерсенна.

4. Задание 2

Реализовать АТД задачи, разработанное в задании 1 практической работы, используя для представления структуры данных современный динамический массив языка C++ -- вектор (vector) – контейнер STL C++.

4.1. Требования к выполнению задания 2

1. Создать копию проекта задания 1 этой практической работы (можно в одном решении среды VS создать новый проект).
2. Реализовать структуру данных АТД на типе struct, с учетом реализации значения множества на векторе из n элементов, выполнив соответствующие изменения в скопированном проекте.
3. Реализовать все операции, определенные в АТД, на реализованной структуре представления данных, корректируя коды функций задания 1, применяя методы контейнера vector при выполнении операций: вставка элемента, удаление элемента (или части вектора), добавления элементов в вектор.
4. Разработать основную программу, демонстрирующую выполнение всех операций над структурой данных на подготовленных тестах. Программа должна управлять всем процессом посредством текстового меню.
5. Выполнить тестирование основной программы на тестах задания 1.

5. Структура отчета

Титульный лист.

Оглавление.

1. Отчет по заданию 1.
 - 1.1. Условие задачи и задание варианта.
 - 1.2. АТД задачи.
 - 1.3. Текст АТД с операциями варианта этой практической работы.

1.4. Разработка программы.

1.4.1. Реализация данных АДТ (представить код структуры).

1.4.2. Алгоритм дополнительной операции варианта из табл. 4.

1.4.3. Таблицы тестов тестирования дополнительной операции варианта.

1.4.4. Код проекта, включающий: файл заголовка, в котором представлена реализация АДТ и его функций, код основной программы.

1.4.5. Скриншоты результатов тестирования.

2. Отчет по заданию 2.

2.1. Коды функций операций вставки, удаления, формирования нового множества заданий 1 и 2, представленные в таблице формата:

Операция	Коды функций задания 1	Коды функций задания 2
Вставить элемент		
Удалить элемент		
Формирование нового множества		

2.2. Код проекта, включающий: файл заголовка, в котором представлена реализация АДТ и его функций; код основной программы.

2.3. Скриншоты результатов тестирования.

6. Контрольные вопросы

1. Что такое указатель?
2. Какие переменные называют динамическими?
3. Как называется участок оперативной памяти, в котором создаются динамические переменные и массивы?
4. Если динамический массив должен быть создан функцией, то как его корректно передать основной программе?
5. Какую структуру называем динамической?
6. Где хранится динамическая структура данных?
7. Где хранятся статические переменные?
8. Когда создается статический массив (структура)?
9. Можно ли увеличить/уменьшить физический объем статического массива во время выполнения программы?
10. Можно ли увеличить/уменьшить физический объем динамического массива во время выполнения программы?
11. Как в программе на языке C++ создать динамическую переменную?
12. Что определяет термин «утечка памяти»?

13. Как устранить утечку памяти в программе на языке C++?
14. Какую структуру данных реализует vector из STL?
15. Что является результатами методов: capacity и size для вектора?
16. Объясните процесс выполнения операции, которую реализует метод push_back вектора.
17. Объясните процесс выполнения операции, которую реализует метод insert вектора.
18. Объясните процесс выполнения операции, которую реализует метод pop_back вектора.
19. Приведите способы реализации операции: *включить новое значение как первый элемент вектора*.
20. Программист написал код наполнения вектора десятью числами:
`vector<int> x; for(int i=0;i<10; i++) std::cin>>x[i];`
При выполнении программы было выброшено исключение при записи значения в элемент вектора. Объясните ошибку программиста.
21. Вектор в коде был определен так: `vector<int> x;` Приведите способы выделения памяти для элементов вектора.
22. Приведите варианты конструкторов вектора, которые определяют его размер.
23. Приведите способы выполнения доступа к отдельному элементу вектора.
24. Объясните назначение и действие функции realloc,
25. Какой метод вектора выполняет те же действия что и функция realloc?
Какой алгоритм реализует функция add? Корректна ли эта функция?

```
void add(int* x,int n) {  
    realloc(x, (n + 1) * sizeof(int));  
    if (x) x[n] = 100;}
```

