

Лабораторная работа № 3 «Работа с массивами данных»

Вариант № 60

Введение

Необходимо спроектировать и реализовать на языке C программу, осуществляющую по запросам пользователя ввод, обработку и вывод последовательности данных, которая представляется в виде массива.

Программа должна реализовывать следующую функциональность:

1. Инициализация массива (заполнение массива в цикле).
2. Вставка нового элемента в массив по индексу.
3. Удаление элемента массива по индексу.
4. Обработка данных (в соответствии с индивидуальным заданием).
5. Вывод текущего состояния массива.

Примечания:

1. Взаимодействие программы с пользователем должно быть выстроено с помощью диалогового меню, имеющего приблизительно следующий вид:
 - (a) Инициализация массива.
 - (b) Вставка нового элемента.
 - (c) Удаление элемента.
 - (d) Индивидуальное задание.
 - (e) Вывод содержимого массива.
2. После выполнения любой из представленных операций программа должна автоматически выводить содержимое массива.
3. Программа должна штатным образом завершаться при выборе пользователем соответствующего пункта диалогового меню или при обнаружении EOF — конца файла (в UNIX-подобных ОС иницируется нажатием клавиш `Ctrl + D`, в Windows — `Ctrl + Z`).
4. Программа должна осуществлять проверку корректности вводимых данных и, в случае ошибок, выдавать соответствующие сообщения, после чего продолжать работу.
5. Обрабатываемые последовательности должны быть представлены в виде массива элементов, которые имеют соответствующий тип данных.

6. Память под массивы обрабатываемых данных должна выделяться и освобождаться динамически, с использованием функций `calloc()`, `malloc()`, `realloc()` и `free()` из состава стандартной библиотеки.
7. Использование VLA (массивов переменной длины) не допускается.
8. При вставке элемента в массив по указанному индексу, элементы массива с индексом, превышающим или совпадающим с заданным, должны «сдвигаться вправо».
9. При удалении элемента массива с указанным индексом, элементы массива с большим индексом должны «сдвигаться влево».
10. При вставке в массив нового элемента по индексу, значение которого превышает максимально допустимое (оно соответствует длине массива), необходимо осуществлять вставку в конец массива.
11. При удалении элемента массива по индексу, значение которого превышает максимально допустимое (оно соответствует длине массива), необходимо вернуть ошибку.
12. Логически законченные части алгоритма решения задачи должны быть оформлены в виде отдельных функций с параметрами. Использование глобальных переменных не допускается.
13. Функции по обработке массивов не должны быть диалоговыми, т. е. они должны принимать все необходимые данные в качестве параметров и возвращать результат работы в виде соответствующих данных и кодов ошибок (исключения: функции инициализации и вывода массивов).
14. Исходные коды программы должны быть логичным образом разбиты на несколько файлов (необходимо использовать как *.c-файлы, так и *.h-файлы).
15. Согласно условиям индивидуального задания, может требоваться наличие нескольких исходных последовательностей. В таком случае, пользователь должен иметь возможность интерактивного взаимодействия с каждой из них.
16. Программа должна корректным образом работать с памятью, для проверки необходимо использовать соответствующие программные средства, например: `valgrind` (при тестировании и отладке программы необходимо запускать её командой вида `valgrind ./lab3`).

Индивидуальное задание

В исходной последовательности найти все числа, в записи которых цифры образуют арифметическую прогрессию при чтении справа налево. Записать данные числа в новую последовательность, удалив их из исходной.

Правила изменения размера выделенной под массив области памяти

Размер выделенной под массив области памяти должен изменяться при выполнении операций, приводящих к изменению количества элементов в массиве.

При заполнении элементами массива всей выделенной под него области памяти её размер должен увеличиваться на объём, необходимый для размещения N дополнительных элементов массива. При наличии в выделенной под массив области памяти места для $N + 1$ новых элементов, её размер должен сократиться на объём, необходимый для хранения N элементов.

В функциях вставки и удаления элементов массива должна быть предусмотрена соответствующая функциональность.