Programming in C

Table of Contents

*1* *Part A* 2

1.1 Prerequisites 2

1.2 Pointer arithmetic on array 2

1.3 Generic array add 3

1.4 Generic array add (part 2) 4

1.5 Mini quiz: pointers & arrays 4

# Part A

## Prerequisites

Please study the modules about arrays.

## Pointer arithmetic on array

Write a small program that declares an array of 5 integers and declare a pointer that points to the last element of the array. Write a loop that traverses the array via the pointer in reverse direction:

int array[] = {10,11,12,13,14};

// Declare a pointer to the last element of array.

for(int i = 0; i < 5; i++);

{

// Print the value at the pointer value

// Set the pointer to the previous array value

}

## Generic array add

I want to have a generic function that can add all values in an array of doubles and return the result. The prototype for this function could be:

double arrayAdd(double\* array);

This prototype has a problem. Please improve the prototype and implement the function. Test if it works properly.

## Generic array add (part 2)

Another way of writing the function in the previous assignment would be by returning the result via a parameter. A reason to do this is if you need to return more than one thing. For example: what would arrayAdd() do if it is called with a NULL pointer? It has no way of telling the caller that an error occurred.

Please improve the arrayAdd() function:

* The function will get a new name: arrayAddParameter()
* Return type is int: it returns 0 if the result is computed correctly, -1 if an error occurs
* The function will get a new parameter named: result. Please define the variable type yourself.

## Generic array (part 3)

Write a function that returns the average of any array.

## Fifo buffer

Implement a fifo buffer that overwrites the oldest element when the buffer is full.  
Write testcases.

Use the function you have developed earlier to calculate the average of the fifo buffer.

Note: please first study the buffer. E.g. see <https://helloacm.com/how-do-you-design-a-circular-fifo-buffer-queue-in-c/>

Note: you can use this function in ES assignments.

## Mini quiz: pointers & arrays

Please try to predict the following questions theoretically. You may, of course, check your answers afterwards in a test program.

The following variables are given:

int a[6] = { 3, 6, -3, 1, 8, 3 };

int i = 3;

int j = 8;

int\* p = a;

int\* q;

int\* r = &a[1];

Please describe if the following statements are correct, and if so what the statement's result is.

1. i = \*(p + 5);
2. j = sizeof (a);
3. j = a[a[a[0]]];
4. r[2] = p[2] + a[1];
5. \*q += \*p \* \*r;