Lab8a

The aim of the task is to create a class Signal representing a signal (time sequence of samples) of any length and a class Mask, necessary for signals processing.

The main function is given (see the main.cpp file) and it should not be modified, except for uncommenting subsequent parts of the task.

Part 1 – 1,5 pts

The Signal class stores the signal samples in a dynamic array (`samples`). Information about the current number of samples in the signal is also stored separately (`no_of_samples`). You must declare and implement in the Signal class the necessary constructor(s)/destructor and other necessary elements. See example usage in main.

Part 2 – 1,5 pts

In the Signal class, the operator + should be implemented, calculating the sum of two signals with the same number of samples (for signals with a different number of samples, an empty signal is returned). Other necessary members should be declared and implemented in the Signal class.

Part 3 – 2 pts

The Mask class stores the filter mask coefficients in a dynamic array (`coeff`). Information about the current number of coefficients (`size`) is also stored separately. The function call operator (operator ()) returns the sum of the filter mask coefficients. Implement all necessary elements. See example usage in main.

Part 4 – 2 pts

In the Signal class, the convolution method should be implemented, which calculates the convolution of the signal with the filter mask.

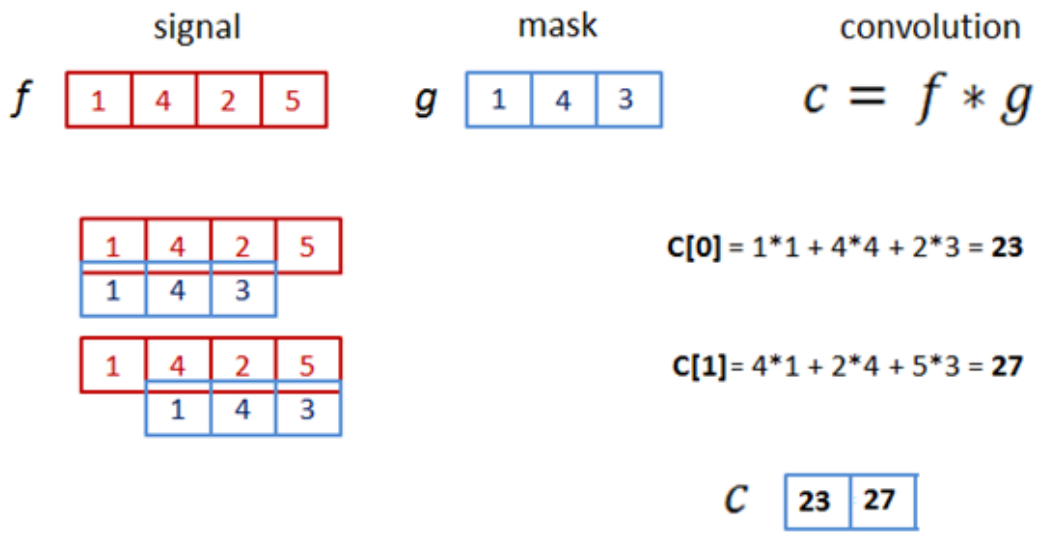
WARNING:
We assume that the filter mask must fit within the signal, so the convolved signal is smaller than the original by the size of the filter mask - 1. For example, the convolution of a signal with 4 samples and a filter with 3 coefficients gives the result of 4 - (3-1) = 2 samples. See the example below.

Then, you should implement linear filtration of the signal with a specific filter mask. Linear filtration assumes signal convolution with a given filter mask with additional normalization of the result by the sum of the coefficients in the mask (divide each sample of the result of the convolution by the sum of the filter mask coefficients).

Part 5 – 1 pts

In the Signal class provide move logic (decelerate and implement move constructor and move assignment).

Example



Example program output:

```
********** Part 1 **********

Signal s0: [ ]
Signal s1: [  0   0   0   0 ]
Signal s2: [  1   2   1   0   1   2   1   0 ]

********** Part 2 **********

Sum s2 + s2: [  2   4   2   0   2   4   2   0 ]
Sum s1 + s2: [ ]

********** Part 3 **********

Mask m0: Mask of size 3 and coefficients [ ]
Mask m1: Mask of size 3 and coefficients [  1   1   1 ]
Sum of coefficints m1: 3

********** Part 4 **********

Convolution s2*m1: [  4   3   2   3   4   3 ]
Linera fitration of s2 with mask m1: [  1.3   1   0.67   1   1.3   1 ]

********** Part 5 **********

With move logic program is still working properly!
```