



## BTH001: Assignment 1

This assignment focus on:

- Composition
  - Deep copying and dynamic memory allocation
    - copy constructor
    - assignment operator
    - destructor
  - Operator overloading
- 

### Register of songs

Your task is to write a program for handling songs. The program will later on be developed further by other system developers, so remember to be clear about the naming and comments in the source code. Attributes for songs are title, artist and play time. Declare and define necessary get- and set functions. A song must also be able to return all of its information as one string. Furthermore the == and != operators must be implemented.

Your system shall be designed from this class sketch:



The class SongRegister shall have a dynamically allocated array of the type pointer to Song (declared as Song\*\*). The initial capacity shall be 5 and the array shall expand if necessary. Copy constructor, destructor and assignment operator shall be implemented – deep copying is required! Keyboard input and screen output in the member functions are not allowed. All songs (Song-objects) must always be placed consecutive in the array.



In the file containing the main function the menu must include the following alternatives:

- Add a new song (user inputs title, artist and play time, use the != or == operator to make sure that no song exists more than once)
- Present all songs with all information
- Present the title of all song with an specific artist (user inputs the artist)
- Present the song (with all information) that has the longest play time and also the song that has the shortest play time
- Remove a song given the title, artist and play time (user inputs these values, use the == operator defined in the class Song while searching for the object to remove)
- Present the total play time (the sum of play times for all songs)

Technical requirements:

- No memory leaks are allowed!  
Use:  
`_CrtSetDbgFlag(_CRTDBG_ALLOC_MEM_DF | _CRTDBG_LEAK_CHECK_DF);`  
at the top of main and run in debug-mode to detect memory leaks.
- Only private member variables are allowed
- All classes shall be divided into h- (declaration) and cpp-files (definition)
- The possibility to implement constant member functions shall be used
- Global variables are not allowed
- The menu based system shall handle separate problems in separate functions