

The following materials have been collected from the numerous sources such as Stanford CS106 and Harvard CS50 including my own and my students over the years of teaching and experiences of programming. Please help me to keep this tutorial up-to-date by reporting any issues or questions. Please send any comments or criticisms to idebtor@gmail.com. Your assistances and comments will be appreciated.

---

## Using Static Library

---

A library is a collection of pre-compiled object files that can be linked into your programs via the linker. Examples are the system functions such as `printf()` and `sqrt()`. There are two types of external libraries: *static library* and *shared library*.

- **A static library** has file extension of **".a"** (archive file) in Unixes or **".lib"** (library) in Windows.
- **A shared library** has file extension of **".so"** (shared objects) in Unixes or **".dll"** (dynamic link library) in Windows. Because of the advantage of dynamic linking, GCC, by default, links to the shared library if it is available. You can list the contents of a library via "nm filename".

### How to build a static library:

---

We use **ar** comand to maintain archive libraries in C/C++. The archive library is a collection of files, typically object files. Using **ar**, you can create a new library, add members to an existing library, delete members from a library, extract members from a library, and print a table of contents for a library.

**Example:** Create a static library, **libsort.a**, that contains **bubbleSort()**, **selectionSort()** and **insertionSort()** functions which are defined in each files(`~/~.cpp`), respectively.

Let's suppose that you have a few source files (**src/bubble.cpp**, **src/selection.cpp**, **src/insertion.cpp**) to turn it into a static library (**lib/libsort.a**).

Assume that you are currently working in **src** folder and keep include files in **~/include/sort.h**.

```
> g++ -c bubble.cpp -o bubble.o
> g++ -c selection.cpp -o selection.o

> ar rcs libsort.a bubble.o selection.o insertion.o
> ar                               // list all the options available
> ar t libsort.a                   // list ~.o files archived
> ar x libsort.a insertion.o // extract ~.o files archived
> ar d libsort.a insertion.o // delete ~.o files archived
> nm bubble.o                    // list the actual function names in .o file

> cp libsort.a ../lib           // copy it and save in lib folder
```

```
ar flags:
r: replace or insert files into archive (with replacement).
c: create an archive file
d: delete files in archive file
s: regenerate the external symbol table
t: display contents of archive
```

**NOTE:** It is important that you recognize that the GCC compiler requires that you prefix your static library with the keyword **lib** and suffix **.a**, like **libsort.a**. The lib prefix is required by the linker to find the static library.

## How to reference a static library

---

Assuming that you are in **~/src** folder and already have **../lib/libsort.a** and **../lib/nowic.a** if necessary.

```
> g++ sortDriver.cpp printList.cpp -o sort -I../include -L../lib -lsort -lnowic
```

```
g++ flags:  -L: Indicates the location of the library you want to reference  
            -l: Specifies the specific library you want to attach
```