



Linux | DB | Open Source | Web

≡ Menu

- [Home](#)
- [Free eBook](#)
- [Start Here](#)
- [Contact](#)
- [About](#)

UNIX ar Examples: How To Create, View, Extract, Modify C Archive Files (*.a)

by Balakrishnan Mariyappan on August 11, 2010

 12  Like 8  Tweet

ar is an archive tool used to combine objects to create an archive file with .a extension, also known as library.

In this article, let us discuss about how to create an user defined static library in C programming using the “ar” utility. The examples shows how to create, extract, and modify the archives using Linux ar command.

To demonstrate the static library creation, let us create two C programs — addition.c and multiplication.c

Using gcc, the object code for these programs are obtained, and the static library libarith.a is created from these two objects.

1. Create Two Sample C Programs

Create addition.c program as shown below.

```
int addition(int a,int b)
{
    int result;
    result = a + b;
    return result;
}
```

Create multiplication.c program as shown below.

```
int multiplication(int a, int b)
{
    int result;
    result = a * b;
    return result;
}
```

A while back we discussed about fundamental of writing C program using [C hello world example](#).

2. Compile the Programs and Get Object Codes

Use -c option to compile both the c program. Using option -c will create the corresponding .o files.

```
$ gcc -c addition.c
```

```
$ gcc -c multiplication.c
```

Now, the current working directory contains both the .c and .o files as shown below.

```
$ ls
addition.c  multiplication.c  addition.o  multiplication.o
```

3. Create the C Program Static Library using ar utility

Now create the static library “libarith.a” with the addition object file and multiplication object file as follows,

```
$ ar cr libarith.a addition.o multiplication.o
```

4. Write C program to Use the Library libarith.a

The library file libarith.a is now ready to usage. Following example indicates how to write a sample C program with the header file to use the libarith.a static library.

Create header.h :

```
#include <stdio.h>
int addition(int a,int b);
int multiplication(int a,int b);
```

Create example.c :

```
#include "header.h"
int main()
{
    int result;
    result = addition(1,2);
    printf("addition result is : %d\n",result);
    result = multiplication(3,2);
    printf("multiplication result is :  %d\n",result);
}
```

Note: [How to Debug C Program using gdb in 5 Simple Steps](http://www.thegeekstuff.com/2010/08/ar-command-examples/) provides step-by-step instruction on debugging your C code.

Compile example.c :

```
$ gcc -Wall example.c -L/home/guest/ -larith -o example
```

The option -L instructs the compiler to look in the /home/guest directory for library files. From this directory, the compiler takes the libarith library file, compiles it with example.c program.

Another method to Compile example.c :

```
$ gcc -Wall example.c libarith.a -o example
```

Execute example executable :

```
$ ./example
addition result is : 3
multiplication result is : 6
```

5. View Object Files in an Archive Using ar Command, option t

To list the object files available in the libarith.a:

```
$ ar t libarith.a  
addition.o  
multiplication.o
```

The options in ar command are similar to the [tar command](#).

6. Extract Object Files from an Archive Using ar Command, option x

You can extract the object files available in an archive as follows.

```
$ mkdir object  
  
$ cp libarith.a object/  
  
$ cd object  
  
$ ar x libarith.a  
  
$ ls *.o  
addition.o  
multiplication.o
```

7. Add an Object File into the Existing Archive Using ar, option r

Let assume that you have create another object file called subtraction.o

The following command extends the libarith.a library file, by inserting subtraction.o object as shown below.

```
$ ar r libarith.a subtraction.o  
  
$ ar t libarith.a  
addition.o  
multiplication.o  
subtraction.o
```

While inserting a .o file, if it already exists in the archive, it would be replaced. Without checking for replacements the objects can be added to end of the archive by using -q option.

8. Delete a Specific Archive Member Using ar, option d

In order to delete a specific archive member from the library file, do the following.

```
$ ar d libarith.a addition.o  
  
$ ar t libarith.a  
multiplication.o  
subtraction.o
```



12

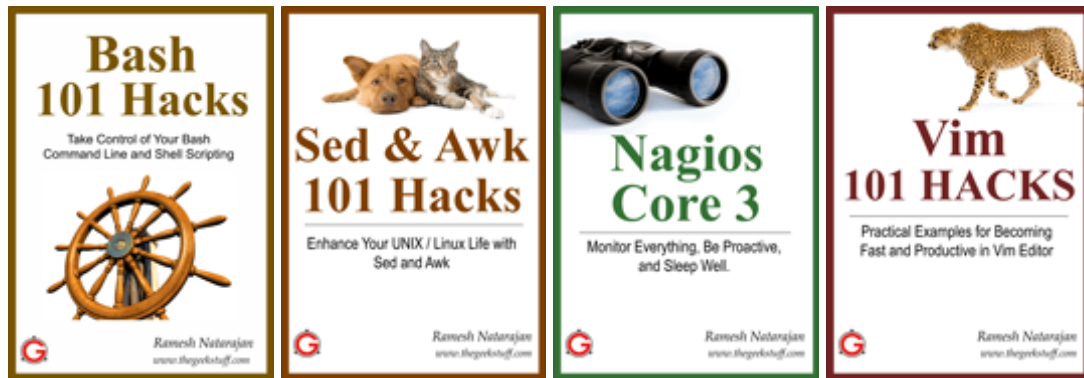
Tweet

Like 8

> [Add your comment](#)

If you enjoyed this article, you might also like..

1. [50 Linux Sysadmin Tutorials](#)
2. [50 Most Frequently Used Linux Commands \(With Examples\)](#)
3. [Top 25 Best Linux Performance Monitoring and Debugging Tools](#)
4. [Mommy, I found it! – 15 Practical Linux Find Command Examples](#)
5. [Linux 101 Hacks 2nd Edition eBook](#) **Free**
- [Awk Introduction – 7 Awk Print Examples](#)
- [Advanced Sed Substitution Examples](#)
- [8 Essential Vim Editor Navigation Fundamentals](#)
- [25 Most Frequently Used Linux IPTables Rules Examples](#)
- [Turbocharge PuTTY with 12 Powerful Add-Ons](#)



Tagged as: [C Program](#)

{ 12 comments... [add one](#) }

- Plip August 11, 2010, 1:40 pm

Hello,

I love your site and have been following it for some time with my rss reader. 😊

But I have a question : why do you write “./file” in some of your commands when “file” would be sufficient ? Is that intended to avoid taking ambiguous symlinks as files ?

(here for example : “\$ gcc -Wall ./example.c ./libarith.a -o ./example”, ./ is useless, as far as I know)

I guess this could lead people following actions word for word to some “bad practices” in that they may think it is necessary to put “./” and dangerous not to write it ...

Anyway, thanks for your work. 😊

Regards,

plip

[Link](#)

- Daniel Reimann August 12, 2010, 4:52 am

Absolutely awesome. Thank you.

[Link](#)

- Solomon Homicz August 17, 2010, 9:53 pm

Would this be fundamentally the same with C++?

Just replace gcc with g++ and .c with .cpp?

Could you also do this with fortran? gfortran with .f90 and .F95?

It would interesting if this could be another way to use c or C++ with fortran subroutines.

[Link](#)

- Ramesh Natarajan August 17, 2010, 11:52 pm

@Plip,

Thanks for the suggestion. I completely agree with you. “./” is not required. It is just a mistake and I’ve removed it from the article.

[Link](#)

- Solomon Homicz August 19, 2010, 11:48 pm

Yep, works fine with fortran subroutines. I used a C++ main file, in a header declared an external C struct matching a common block in a fortran header and declared an external C function (void). Compiled the fortran as an object file used ar to make a .a and then compiled with g++, works great! I could write a C++ (or C) wrapper for my fortran subroutines package it up in a library and have an object I could just plug in and use in C++. Way cool!

[Link](#)

- Mars December 14, 2010, 11:32 pm

informative article
thanks

[Link](#)

- boris May 10, 2012, 11:00 pm

Hi
It doesn't work
I get /usr/bin/ld: cannot find -larith.a

Which way to repair it

[Link](#)

- Jonas Kulla August 17, 2012, 3:44 pm

Thank you very much!
Even though I have been programming for quite a while in C,
I really appreciated detailed / very simply step-by-steps like these,
much better than skimming through manuals..

[Link](#)

- Gagandeep Bali February 27, 2013, 6:01 am

Wonderful stuff, helped me loads, learning the basics for creating library files and adding my own functions to it. Really full of knowledge. Nice endeavour of yours, will always appreciate what I learned on this blog of yours 😊

[Link](#)

- Anonymous August 9, 2014, 6:37 am

I think you forgot the -s option for indexing the archive, because I can't link with the library (statically), because it says that the archive has no index (therefore required). So instead of "ar -cr" use "ar -crs"

[Link](#)

- sravani January 6, 2016, 6:31 am

Hi,
Very useful information in detailed way.
I was able to get .o files from an archive file extension .a
Can anybody please let me know is there any tool or set of commands to do the reverse engineering (conversion from .a to c files or to any readable format).
Thank you.

[Link](#)

- vismitha February 18, 2016, 10:02 am

Clear notes

[Link](#)

Leave a Comment

Name

Email

Website

Comment

☐ Notify me of followup comments via e-mail

Next post: [How to Allow MySQL Client to Connect to Remote MySQL server](#)

Previous post: [RAID 0, RAID 1, RAID 5, RAID 10 Explained with Diagrams](#)

[RSS](#) | [Email](#) | [Twitter](#) | [Facebook](#) | [Google+](#)

EBOOKS

- **Free** [Linux 101 Hacks 2nd Edition eBook](#) - Practical Examples to Build a Strong Foundation in Linux
- [Bash 101 Hacks eBook](#) - Take Control of Your Bash Command Line and Shell Scripting
- [Sed and Awk 101 Hacks eBook](#) - Enhance Your UNIX / Linux Life with Sed and Awk
- [Vim 101 Hacks eBook](#) - Practical Examples for Becoming Fast and Productive in Vim Editor
- [Nagios Core 3 eBook](#) - Monitor Everything, Be Proactive, and Sleep Well



The Geek Stuff

16,599 likes

Be the first of your friends to like this



POPULAR POSTS

- [12 Amazing and Essential Linux Books To Enrich Your Brain and Library](#)
- [50 UNIX / Linux Sysadmin Tutorials](#)
- [50 Most Frequently Used UNIX / Linux Commands \(With Examples\)](#)
- [How To Be Productive and Get Things Done Using GTD](#)

- [30 Things To Do When you are Bored and have a Computer](#)
- [Linux Directory Structure \(File System Structure\) Explained with Examples](#)
- [Linux Crontab: 15 Awesome Cron Job Examples](#)
- [Get a Grip on the Grep! – 15 Practical Grep Command Examples](#)
- [Unix LS Command: 15 Practical Examples](#)
- [15 Examples To Master Linux Command Line History](#)
- [Top 10 Open Source Bug Tracking System](#)
- [Vi and Vim Macro Tutorial: How To Record and Play](#)
- [Mommy, I found it! -- 15 Practical Linux Find Command Examples](#)
- [15 Awesome Gmail Tips and Tricks](#)
- [15 Awesome Google Search Tips and Tricks](#)
- [RAID 0, RAID 1, RAID 5, RAID 10 Explained with Diagrams](#)
- [Can You Top This? 15 Practical Linux Top Command Examples](#)
- [Top 5 Best System Monitoring Tools](#)
- [Top 5 Best Linux OS Distributions](#)
- [How To Monitor Remote Linux Host using Nagios 3.0](#)
- [Awk Introduction Tutorial – 7 Awk Print Examples](#)
- [How to Backup Linux? 15 rsync Command Examples](#)
- [The Ultimate Wget Download Guide With 15 Awesome Examples](#)
- [Top 5 Best Linux Text Editors](#)
- [Packet Analyzer: 15 TCPDUMP Command Examples](#)
- [The Ultimate Bash Array Tutorial with 15 Examples](#)
- [3 Steps to Perform SSH Login Without Password Using ssh-keygen & ssh-copy-id](#)
- [Unix Sed Tutorial: Advanced Sed Substitution Examples](#)
- [UNIX / Linux: 10 Netstat Command Examples](#)
- [The Ultimate Guide for Creating Strong Passwords](#)
- [6 Steps to Secure Your Home Wireless Network](#)
- [Turbocharge PuTTY with 12 Powerful Add-Ons](#)

CATEGORIES

- [Linux Tutorials](#)
- [Vim Editor](#)
- [Sed Scripting](#)
- [Awk Scripting](#)
- [Bash Shell Scripting](#)
- [Nagios Monitoring](#)
- [OpenSSH](#)
- [IPTables Firewall](#)
- [Apache Web Server](#)
- [MySQL Database](#)
- [Perl Programming](#)
- [Google Tutorials](#)
- [Ubuntu Tutorials](#)
- [PostgreSQL DB](#)
- [Hello World Examples](#)
- [C Programming](#)
- [C++ Programming](#)
- [DELL Server Tutorials](#)
- [Oracle Database](#)
- [VMware Tutorials](#)

Ramesh Natarajan



About The Geek Stuff



My name is **Ramesh Natarajan**. I will be posting instruction guides, how-to, troubleshooting tips and tricks on Linux, database, hardware, security and web. My focus is to write articles that will either teach you or help you resolve a problem. Read more about [Ramesh Natarajan](#) and the blog.

Contact Us

Email Me : Use this [Contact Form](#) to get in touch me with your comments, questions or suggestions about this site. You can also simply drop me a line to say hello!.

[Follow us on Google+](#)

[Follow us on Twitter](#)

[Become a fan on Facebook](#)

Support Us

Support this blog by purchasing one of my ebooks.

[Bash 101 Hacks eBook](#)

[Sed and Awk 101 Hacks eBook](#)

[Vim 101 Hacks eBook](#)

[Nagios Core 3 eBook](#)

Copyright © 2008–2017 Ramesh Natarajan. All rights reserved | [Terms of Service](#)