

Libraries i C

— En kort introduktion

Thomas Bøgholm

Aalborg Universitet

Marts 2010

Outline

- Motivation
- Hvad er et library
- Brugen af libraries
- Eksempler - med live kode
- Dit eget library
- Opgaver

Motivation

- Modul lignende software udvikling
- Programmering idag er ca. kun brug af libraries og APIer
 - det er sjældent man har brug for at lave noget fra bunden
- Ofte er folk usikre på hvordan libraries bruges
- Miniprojekt handler om kernemoduler...
 - Modning af C kundskaber
 - Det minder om kernel modules (er ca. det omvendte)

Ikke rocket-science

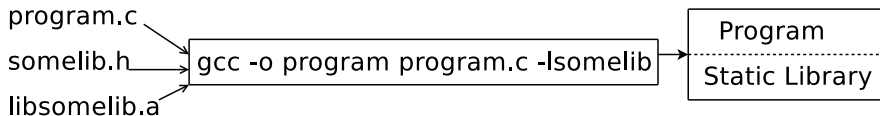
- Det handler om at læse dokumentation
 - eller klare sig med hvad man kan finde
- Det handler om at kunne læse/skrive C
 - ved mangel af dokumentation
 - det ser ofte farligt ud med pointere, datastrukturer osv.

Libraries

- Et library består af:
 - headers `.h` - indeholdende diverse erklæringer
 - objektfil `.a` / `.so` - selve koden
- Til brug i egne programmer
- Static libraries
- Shared libraries
- Det handler om at finde/bruge (alternativer til?) dokumentation
 - Man skal ikke være bange for at åbne header filer
 - Ofte er andres kode interessant:
 - code.google.com, `apt-get source`, etc

Static libraries

- Kendt på compiletime
- “Kopieres” ind i executables
 - man kan evt. nøjes med den brugte delmængde
- ikke delt
- mindre flexibilitet



Shared libraries

- Brugt på runtime
- Kan skiftes ud og opdateres
- Delt mellem flere processer
- Højere flexibilitet



Eksempler på libraries

- math
- pthreads
- bluetooth/BlueZ
- opengl
- etc etc

- Statisk linking
 - libXX.a – gcc -static source.c -o target -lXX
- Dynamisk linking
 - libYY.so – gcc source.c -o target -lYY
- Manuel loading af libraries
 - dlfcn.h - interface til dynamic linking loader
 - dlopen()
 - dlsym()
 - dlclose()
 - dlerror()

Tråde i C – pthreads

- pthreads er et tråd-library
- Implementeret i libpthread
- Demo

Dit eget library

- Der findes flere måder at lave/bruge moduler på
- Statiske libraries
- Dynamiske libraries
 - med automatisk loading
 - med manuel loading

Static library

- prefix: lib
- suffix: .a (archive)
- Samling ikke-linkede objekt-filer
 - gcc bruges som normalt (-c)
 - ar bruges til at samle et antal .o filer
 - den resulterende .a fil installeres
 - normal kompilering med gcc -l option
- Eksempel

Shared library

- prefix: lib
- suffix: .so (shared object)
- En samling position independent objekt-filer
 - gcc skal nu generere “position independent code” (-fPIC)
 - et antal .o filer samles med gcc og en masse options
 - gcc skal lave et shared object: -shared
 - linkerens navngiver med soname: -Wl,-soname,libAwesome.so.0
 - den resulterende .so fil installeres
 - normal kompilering med gcc -l option
- Eksempel

Manuel loading af libraries

- Normalt vil et shared library loades automatisk
- Manglende library?
- Løsning: manuel loading
- Brug af libraries foregår da via funktionspointere
 - dlfcn.h - interface til dynamic linking loader
 - dlopen()
 - dlsym()
 - dlclose()
 - dlerror()
- kompilering med -ldl
- Eksempel

C programmering:

- Gør det til en vane at:
 - Læse dokumentation
 - Læse kode
 - frygt aldrig at åbne koden!
- Med lidt tålmodighed er det:
 - Sjovt at kode C
 - og ikke svært
 - Det er let at bruge libraries
 - og endda at lave libraries

Links

- <http://www.dwheeler.com/program-library/>
- <http://www.yolinux.com/TUTORIALS/LibraryArchives-StaticAndDynamic.html>
- <http://www.cs.aau.dk/~boegholm/libslides.pdf>
- manpages-dev
- apt-get source PAKKE
- /usr/include

Målet med disse opgaver

- At kaste jer ud på dybt vand
- I får brug for at læse dokumentation
- I får brug for at prøve jer frem
- Ved kerneudvikling (og generelt?) vil I bruge mest tid på at læse dokumentation
- Derfor er opgaverne ret åbne

Brug libraries!

- 1
 - Lav racekondition eksempel med pthreads
 - Indsæt synkronisering
 - pthread_mutex_t
 - sem_t (semaphore.h)
- 2
 - Simpelt plugin system
 - Lav et shared library og brug det med funktionerne:
 - dlopen()
 - dlsym()
 - dlclose()
- 3
 - Hvis du har et bluetooth device
 - brug libbluetooth
 - Udskriv en liste af de devices der findes
 - bluetooth/hci.h
 - bluetooth/bluetooth.h