2021_11_25

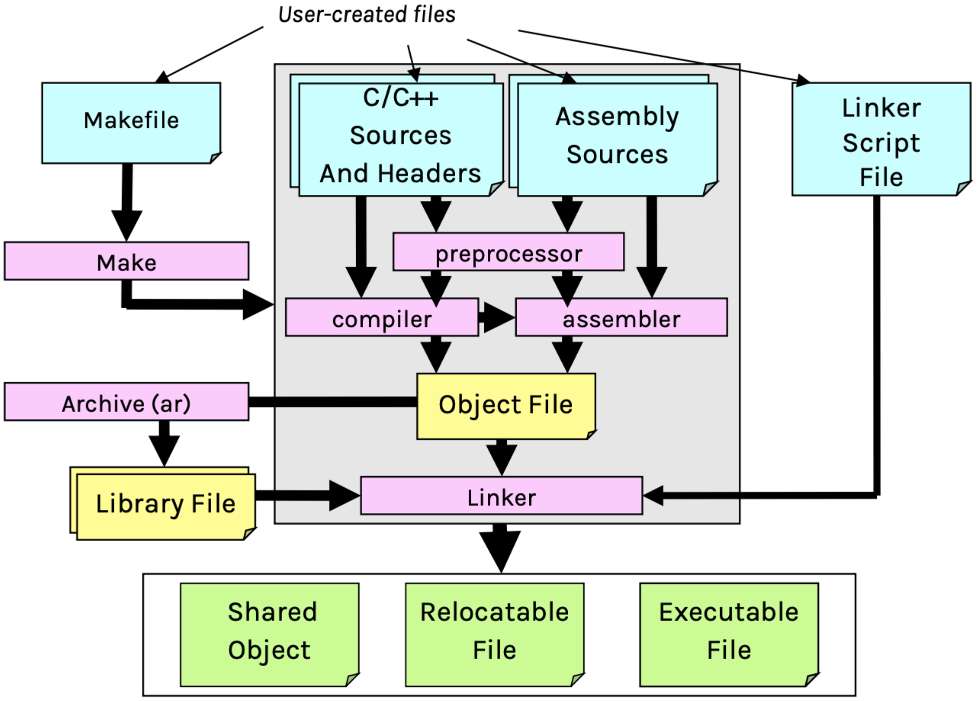
November 26, 2021

1 2021-11-25

1.0.1 Corso ITS

1.1 Magento & e-commerce software

1.2 ### Fondamenti di Programmazione (Andrea Ribuoli)



Su Google: “kernighan ritchie linguaggio c pdf”

Invocando *make* in assenza di Makefile

```
[5]: !make
```

```
make: *** No targets specified and no makefile found.  Stop.
```

Invocando *make* in presenza di Makefile vuoto

[6]:

```
!make
```

```
make: *** No targets.  Stop.
```

Invocando *make* in presenza di un Makefile apparentemente valido ma mancante di TAB Ogni azione specificata dopo la regola di dipendenza deve essere *tab-indented*, sfortunatamente alcuni editor trasformano il carattere di controllo **TAB** (`\t`) in un numero di spazi.

[11]:

```
!make -n
```

```
Makefile:2: *** missing separator.  Stop.
```

[1]:

```
!hexdump -c Makefile
```

```
0000000   a   l   l       :       c   y   r   i   l       a   s   c   i
0000010   i  \n  \n   c   y   r   i   l       :       t   a   b   e   l
0000020   l   a   _   c   y   r   i   l   l   i   c   .   o       c   o
0000030   n   v   e   r   t   i   _   a   p   i   .   o  \n  \t   g   c
0000040   c       -   o       c   y   r   i   l       t   a   b   e   l
0000050   l   a   _   c   y   r   i   l   l   i   c   .   o       c   o
0000060   n   v   e   r   t   i   _   a   p   i   .   o  \n
0000070      \n   a   s   c   i   i       :       t   a   b   e   l   l
0000080   a   _   a   s   c   i   i   .   o       c   o   n   v   e   r
0000090   t   i   _   a   p   i   .   o                      \n  \t   g
00000a0   c   c       -   o       a   s   c   i   i       t   a   b   e
00000b0   l   l   a   _   a   s   c   i   i   .   o       c   o   n   v
00000c0   e   r   t   i   _   a   p   i   .   o  \n                  \n
00000d0   t   a   b   e   l   l   a   _   a   s   c   i   i   .   o
00000e0   :       t   a   b   e   l   l   a   _   a   s   c   i   i   .
00000f0   c       c   o   n   v   e   r   t   i   _   a   p   i   .   h
0000100  \n  \t   g   c   c       -   c       -   o       t   a   b   e
0000110   l   l   a   _   a   s   c   i   i   .   o       t   a   b   e
0000120   l   l   a   _   a   s   c   i   i   .   c  \n
0000130  \n   t   a   b   e   l   l   a   _   c   y   r   i   l   l   i
0000140   c   .   o       :       t   a   b   e   l   l   a   _   c   y
0000150   r   i   l   l   i   c   .   c       c   o   n   v   e   r   t
0000160   i   _   a   p   i   .   h  \n  \t   g   c   c       -   c
0000170   -   o       t   a   b   e   l   l   a   _   c   y   r   i   l
0000180   l   i   c   .   o       t   a   b   e   l   l   a   _   c   y
0000190   r   i   l   l   i   c   .   c  \n                  \n   c   o
00001a0   n   v   e   r   t   i   _   a   p   i   .   o       :       c
00001b0   o   n   v   e   r   t   i   _   a   p   i   .   c       c   o
00001c0   n   v   e   r   t   i   _   a   p   i   .   h  \n  \t   g   c
00001d0   c       -   c       -   o       c   o   n   v   e   r   t   i
00001e0   _   a   p   i   .   o       c   o   n   v   e   r   t   i   _
00001f0   a   p   i   .   c  \n
00001f6
```

1.3 Makefile

```
all : cyril ascii

cyril : tabella_cyrillic.o converti_api.o
	gcc -o cyril tabella_cyrillic.o converti_api.o

ascii : tabella_ascii.o converti_api.o
	gcc -o ascii tabella_ascii.o converti_api.o

tabella_ascii.o : tabella_ascii.c converti_api.h
	gcc -c -o tabella_ascii.o tabella_ascii.c

tabella_cyrillic.o : tabella_cyrillic.c converti_api.h
	gcc -c -o tabella_cyrillic.o tabella_cyrillic.c

converti_api.o : converti_api.c converti_api.h
	gcc -c -o converti_api.o converti_api.c
```

[2]:

```
!make ascii -n
```

```
gcc -c -o tabella_ascii.o tabella_ascii.c
gcc -c -o converti_api.o converti_api.c
gcc -o ascii tabella_ascii.o converti_api.o
```

[3]:

```
!make cyril -n
```

```
gcc -c -o tabella_cyrillic.o tabella_cyrillic.c
gcc -c -o converti_api.o converti_api.c
gcc -o cyril tabella_cyrillic.o converti_api.o
```

[4]:

```
!make ascii
```

```
gcc -c -o tabella_ascii.o tabella_ascii.c
gcc -c -o converti_api.o converti_api.c
gcc -o ascii tabella_ascii.o converti_api.o
```

[5]:

```
!make ascii -n
```

```
make: 'ascii' is up to date.
```

[6]:

```
!make cyril -n
```

```
gcc -c -o tabella_cyrillic.o tabella_cyrillic.c
gcc -o cyril tabella_cyrillic.o converti_api.o
```

[8]:

```
!make cyril
```

```
gcc -c -o tabella_cyrillic.o tabella_cyrillic.c
gcc -o cyril tabella_cyrillic.o converti_api.o
```

[9]:
```
!make -n
```

```
make: Nothing to be done for 'all'.
```

[11]:
```
!gcc --help
```

```
Usage: gcc-6.bin [options] file…
Options:
  -pass-exit-codes         Exit with highest error code from a phase.
  --help                   Display this information.
  --target-help            Display target specific command line options.
  --help={common|optimizers|params|target|warnings|[^]{joined|separate|undocumen
ted}}[,…].
                           Display specific types of command line options.
  (Use '-v --help' to display command line options of sub-processes).
  --version                Display compiler version information.
  -dumpspecs               Display all of the built in spec strings.
  -dumpversion             Display the version of the compiler.
  -dumpmachine             Display the compiler's target processor.
  -print-search-dirs       Display the directories in the compiler's search
path.
  -print-libgcc-file-name  Display the name of the compiler's companion library.
  -print-file-name=<lib>   Display the full path to library <lib>.
  -print-prog-name=<prog>  Display the full path to compiler component <prog>.
  -print-multiarch         Display the target's normalized GNU triplet, used as
                           a component in the library path.
  -print-multi-directory   Display the root directory for versions of libgcc.
  -print-multi-lib         Display the mapping between command line options and
                           multiple library search directories.
  -print-multi-os-directory Display the relative path to OS libraries.
  -print-sysroot           Display the target libraries directory.
  -print-sysroot-headers-suffix Display the sysroot suffix used to find headers.
  -Wa,<options>            Pass comma-separated <options> on to the assembler.
  -Wp,<options>            Pass comma-separated <options> on to the
preprocessor.
  -Wl,<options>            Pass comma-separated <options> on to the linker.
  -Xassembler <arg>        Pass <arg> on to the assembler.
  -Xpreprocessor <arg>     Pass <arg> on to the preprocessor.
  -Xlinker <arg>           Pass <arg> on to the linker.
  -save-temps              Do not delete intermediate files.
  -save-temps=<arg>        Do not delete intermediate files.
  -no-canonical-prefixes   Do not canonicalize paths when building relative
                           prefixes to other gcc components.
  -pipe                    Use pipes rather than intermediate files.
  -time                    Time the execution of each subprocess.
  -specs=<file>            Override built-in specs with the contents of <file>.
  -std=<standard>          Assume that the input sources are for <standard>.
  --sysroot=<directory>    Use <directory> as the root directory for headers
```

```
                           and libraries.
  -B <directory>           Add <directory> to the compiler's search paths.
  -v                       Display the programs invoked by the compiler.
  -###                     Like -v but options quoted and commands not executed.
  -E                       Preprocess only; do not compile, assemble or link.
  -S                       Compile only; do not assemble or link.
  -c                       Compile and assemble, but do not link.
  -o <file>                Place the output into <file>.
  -pie                     Create a position independent executable.
  -shared                  Create a shared library.
  -x <language>            Specify the language of the following input files.
                           Permissible languages include: c c++ assembler none
                           'none' means revert to the default behavior of
                           guessing the language based on the file's extension.

Options starting with -g, -f, -m, -O, -W, or --param are automatically
 passed on to the various sub-processes invoked by gcc-6.bin.  In order to pass
 other options on to these processes the -W<letter> options must be used.

For bug reporting instructions, please see:
<http://gcc.gnu.org/bugs.html>.
```

[12]:
```
!make --help
```

```
Usage: make [options] [target] …
Options:
  -b, -m                      Ignored for compatibility.
  -B, --always-make           Unconditionally make all targets.
  -C DIRECTORY, --directory=DIRECTORY
                              Change to DIRECTORY before doing anything.
  -d                          Print lots of debugging information.
  --debug[=FLAGS]             Print various types of debugging information.
  -e, --environment-overrides
                              Environment variables override makefiles.
  --eval=STRING               Evaluate STRING as a makefile statement.
  -f FILE, --file=FILE, --makefile=FILE
                              Read FILE as a makefile.
  -h, --help                  Print this message and exit.
  -i, --ignore-errors         Ignore errors from recipes.
  -I DIRECTORY, --include-dir=DIRECTORY
                              Search DIRECTORY for included makefiles.
  -j [N], --jobs[=N]          Allow N jobs at once; infinite jobs with no arg.
  -k, --keep-going            Keep going when some targets can't be made.
  -l [N], --load-average[=N], --max-load[=N]
                              Don't start multiple jobs unless load is below N.
  -L, --check-symlink-times   Use the latest mtime between symlinks and target.
  -n, --just-print, --dry-run, --recon
                              Don't actually run any recipe; just print them.
  -o FILE, --old-file=FILE, --assume-old=FILE
                              Consider FILE to be very old and don't remake it.
  -O[TYPE], --output-sync[=TYPE]
                              Synchronize output of parallel jobs by TYPE.
  -p, --print-data-base       Print make's internal database.
  -q, --question              Run no recipe; exit status says if up to date.
  -r, --no-builtin-rules      Disable the built-in implicit rules.
  -R, --no-builtin-variables  Disable the built-in variable settings.
  -s, --silent, --quiet       Don't echo recipes.
  -S, --no-keep-going, --stop
                              Turns off -k.
  -t, --touch                 Touch targets instead of remaking them.
  --trace                     Print tracing information.
  -v, --version               Print the version number of make and exit.
  -w, --print-directory       Print the current directory.
  --no-print-directory        Turn off -w, even if it was turned on implicitly.
  -W FILE, --what-if=FILE, --new-file=FILE, --assume-new=FILE
                              Consider FILE to be infinitely new.
  --warn-undefined-variables  Warn when an undefined variable is referenced.

This program built for powerpc-ibm-os400
Report bugs to <bug-make@gnu.org>
```

[43]:

```
!gcc -o cyril tabella_cyrillic.o
```

```
ld: 0711-317 ERRORE: Simbolo non definito : .converti_api
ld: 0711-345 Usare l'opzione -bloadmap o -bnoquiet per ulteriori informazioni.
collect2: error: ld returned 8 exit status
```

La risoluzione dei simboli (o chiamate di funzione) esterne può avvenire, ancora in fase di costruzione, anche tramite l'utilizzo di librerie. Approfondiremo il tema in una prossima lezione.

1.3.1 Diverso il caso delle cosiddette SHARED LIBRARY (o DLL)

Se il sistema operativo le supporta offrono la possibilità di differimento della risoluzione dei riferimenti esterni alla fase di esecuzione del programma: è la modalità oggi dominante nell'architettare il software complesso.

Abbiamo spiegato il significato di **semantic versioning**. Quando viene adottato ad ogni particoare pacchetto software viene assegnata una terna di valori numerici separati da punti: ad esempio 11.2.0 I tre valori si interpretano come MAJOR.MINOR.PATCH Il codice che è stato sviluppato adottando il componente come pre-requisito non richiede modifiche se l'aggiornamento non modifica il numero MAJOR.