

• MAKEFILE

- Serve per automatizzare ed ottimizzare l'azione di compilazione dei file sorgente (in particolare in un make-file si ignorano i file già compilati).
e definisce in semplice linguaggio per questo scopo.

- Quando eseguiamo il comando make a meno di indicazioni aggiuntive eseguirà nell'ordine (nelle directory corrente)

1. GNU makefile.

2. makefile.

3. Makefile.

ma non dobbiamo chiamare il file makefile o Makefile o GNUmakefile per forza, possiamo creare un make file con

`make -f my-make-file`

- Regole base

la struttura di un file makefile è così:

Target: Prerequisites

<TAB> Recipe

<TAB> Recipe

...

<TAB> Recipe

• dove il target è il nome di un eseguibile (FILE oggetto)

• Prerequisites è un insieme di file .c o .h o altri target

• Recipe è un comando da eseguire

- Un target senza prerequisiti è detto PHONY (FASULLO) ed è come avere indicator come PHONY

ESEMPIO:

.PHONY clean
clean:

rm -f *.o merge-sorted-list

- Nel caso in cui abbia più target, se non specifichiamo nulla viene eseguito il 1° target. solitamente viene aggiunto un target ALL (con prerequisiti tutti i target).

VARIABILI

possiamo nei make file assegnare delle variabili...

ASSEGNAZIONE BASE:

VAR := val

ACCESSO AL CONTENUTO DI UNA VARIABLE:

\$(VAR)

ESPANSIONE COMPLETA

VAR := val

ESPANSIONE TESTUALE

VAR = val

DIFFERENZE:

con l'espansione completa

BAR := Hello

FOO := \$(BAR) world

modifica successiva

BAR := Hi

⇒

FOO = Hello world

con l'espansione testuale

BAR = Hello

FOO = \$(BAR) world

modifica macchina

BAR = Hi

⇒

FOO = Hi world

In sintesi:

- nell'espansione completa il valore delle variabili viene determinato al momento dell'esecuzione.
- nell'espansione testuale il valore delle variabili viene determinato al momento dell'inv.

DEFINIZIONE CONDIZIONALE

VAR ?= val

(ASSEGNATA SOLO SE VAR È vuota)

CONCATENAZIONE

VAR += val

- VARIABILI AUTOMATICHE

\$^: INDICA LA LISTA DEI PREREQUISITI

\$<: PRIMO PREREQUISITO

\$G: TARGET

- VARIABILI IMPLIcite

cc: COMPILATORE

eflags: FLAG PER COMPILAZIONE (-Wall, -Wextra)

ldflags: FLAG PER LINKER

ldlibs ↑