

Editor e compilatore

Marco Alberti

Programmazione e Laboratorio, A.A. 2016-2017

Dipartimento di Matematica e Informatica - Università di Ferrara

Ultima modifica: 4 ottobre 2016

- Qualsiasi programma in grado di creare e modificare file di testo (in formato ASCII o nelle sue estensioni, come ISO-8859-1 e UTF-8)
- Alcuni editor supportano particolarmente bene la scrittura di file sorgenti C e l'integrazione con gli strumenti di programmazione (compilatore, debugger, version control): Emacs, Vim, Sublime Text
- Noi useremo Visual Studio Code (<https://code.visualstudio.com>)
- Estensione (plugin) per lo sviluppo in C documentata qui: <https://code.visualstudio.com/docs/languages/cpp>.

Esercizio

1. Scrivere un programma in linguaggio C che stampi a video la stringa "Hello, World!"
2. Salvarlo nella cartella `~/programmazione/hello_world`

- GNU's Not Unix! (<https://www.gnu.org>)
- GNU Compiler Collection (originariamente, GNU C Compiler) (<https://gcc.gnu.org>)
- Compilatori per C, C++ , altri linguaggi
- Disponibile per quasi tutte le piattaforme
- Molto ben documentato (`gcc -h`, `man gcc`, `info gcc`)

Compilazione

```
gcc __nome__.c
```

compila (e linka) il file **sorgente** C `__nome.c__`, generando il file **eseguibile** `a.out`.

```
gcc -o __eseguibile__ __nome.c__
```

compila (e linka) il file sorgente C `__nome.c__`, generando il file eseguibile `__eseguibile__`.

Compilazione e linking separati

```
gcc -c __nome__.c
```

compila (ma non linka) il file sorgente C `__nome__.c`,
generando il file **oggetto** `__nome__.o`

```
gcc -l__lib__ __nome__.o
```

linka il file oggetto `__nome__.o` con la libreria `lib__lib__`,
generando il file eseguibile `a.out` (è possibile specificare il
nome `__eseguibile__` con l'opzione `-o __eseguibile__`).

Esempio

```
gcc -c hello.c
```

```
gcc -lc -o hello hello.o
```

Compilazione in Assembly e assembler

```
gcc -S __nome__.c
```

compila il file sorgente C `__nome__.c` nel file Assembly `__nome__.s` (con l'opzione `-masm=intel` produce Assembly in sintassi Intel anziché in quella predefinita AT&T).

```
as -o __nome__.o __nome__.s
```

assembla il file `__nome__.s` nel file `__nome__.o` (che dovrà poi essere linkato come nell'esempio precedente).

1. Compilare il file precedentemente creato
2. Eseguire il file eseguibile generato