

Laboratorio di Programmazione

Introduzione alla programmazione in C

Dott.ssa Veronica Rossano
rossano@di.uniba.it
<http://www.di.uniba.it/~rossano>

Testi adottati

- A. Kelley, I. Pohl
C Didattica e Programmazione
Pearson Education/Addison-Wesley
- B.W. Kernighan, D.M. Ritchie
Linguaggio C (ANSI C), II ed.
Pearson Education/Addison-Wesley

Testi consigliati

- H.M. Deitel, P.J. Deitel, *Corso Completo di programmazione (Terza Edizione)*, APOGEO EDUCATION
- Hanly J.R., Koffman E.B.
Problem Solving and Program Design in C
Pearson Education/Addison-Wesley

Dispense

- Le dispense del laboratorio di programmazione sono disponibili sulla piattaforma di e-learning all'indirizzo:
<http://multimedialab.di.uniba.it/>

Ambiente di sviluppo...

- Eclipse IDE for C/C++

Eclipse is an open source community whose projects are focused on building an extensible development platform, runtimes and application frameworks for building, deploying and managing software across the entire software lifecycle. Many people know us, and hopefully love us, as a Java IDE but Eclipse is much more than a Java IDE.

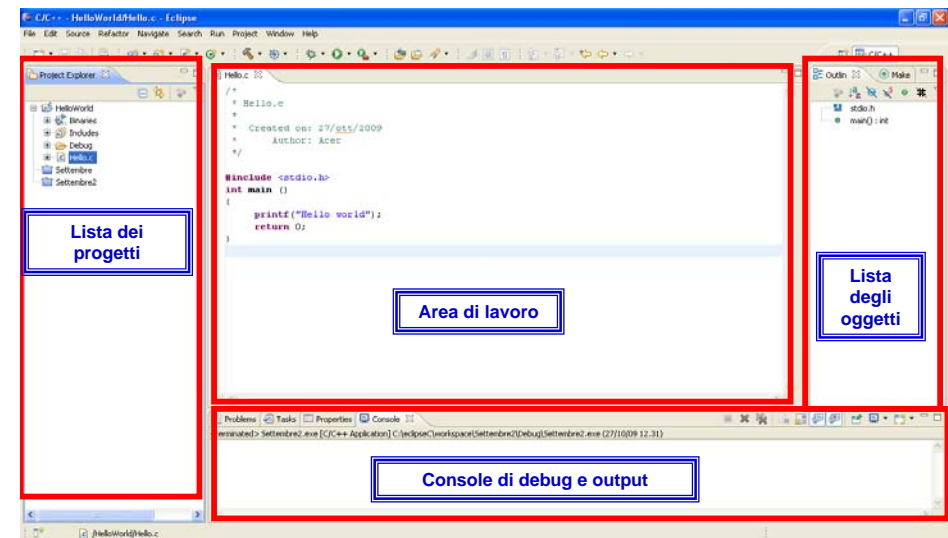
...Ambiente di sviluppo...

- Per scaricare l'IDE di Eclipse
<http://www.eclipse.org/downloads/>
- Oltre all'ambiente di sviluppo è necessario installare anche il compilatore C
- Le istruzioni di installazione dell'IDE di Eclipse e del compilatore **MinGW** sono reperibili all'indirizzo
http://maresca.dis.unina.it/wiki/index.php/Installazione_di_Eclipse

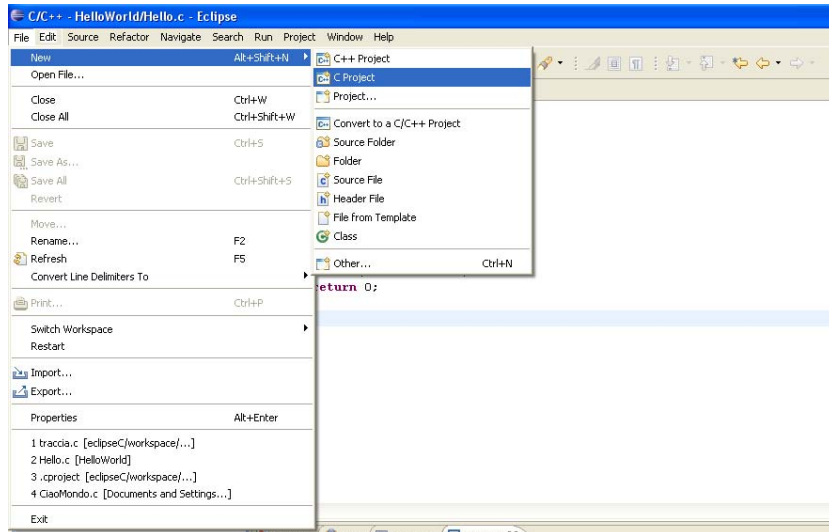
...Ambiente di sviluppo (in alternativa)

- Dev-C++ ambiente integrato di sviluppo per linguaggi di programmazione C/C++
- Dev-C++ è un Free Software distribuito con licenza GNU General Public License (GPL)
- L'ambiente è facilmente reperibile sulla rete, ed inoltre è scaricabile dalla piattaforma di e-learning

Il workbench di ECLIPSE



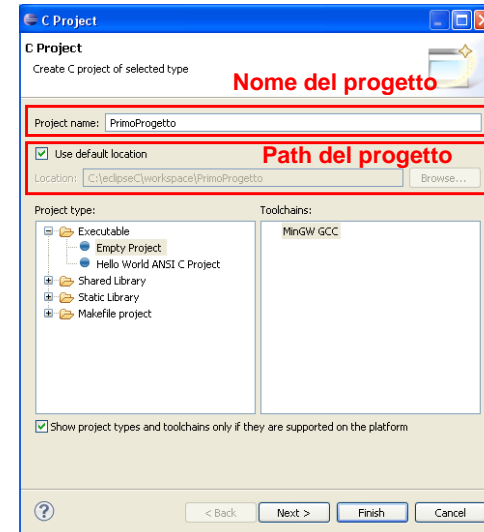
Eclipse: creare un progetto...



Corso laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

9/43

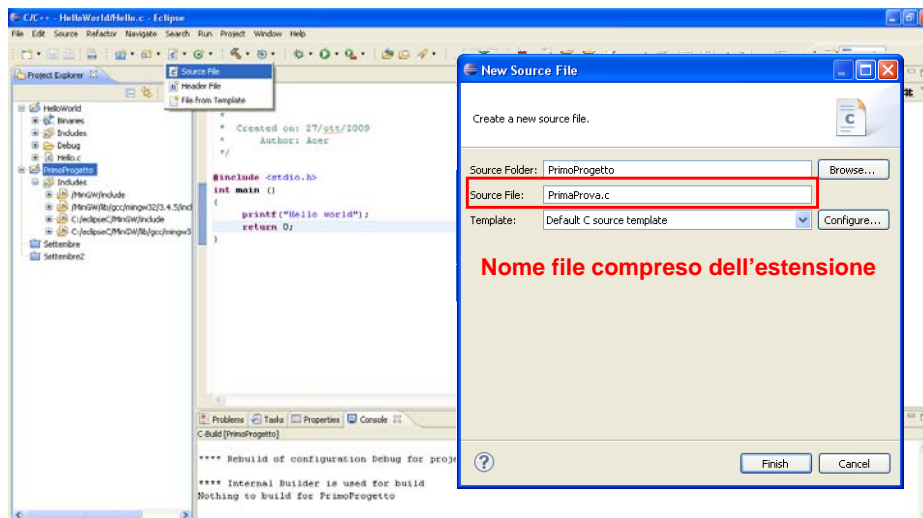
...Eclipse: creare un progetto



Corso laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

10/43

...Eclipse: creare un file C



Il primo programma in C...

```
#include <stdio.h>

int main ()
{
    printf("Hello World!");
    return 0;
}
```

- È un semplice programma che stampa a video la stringa Hello World!

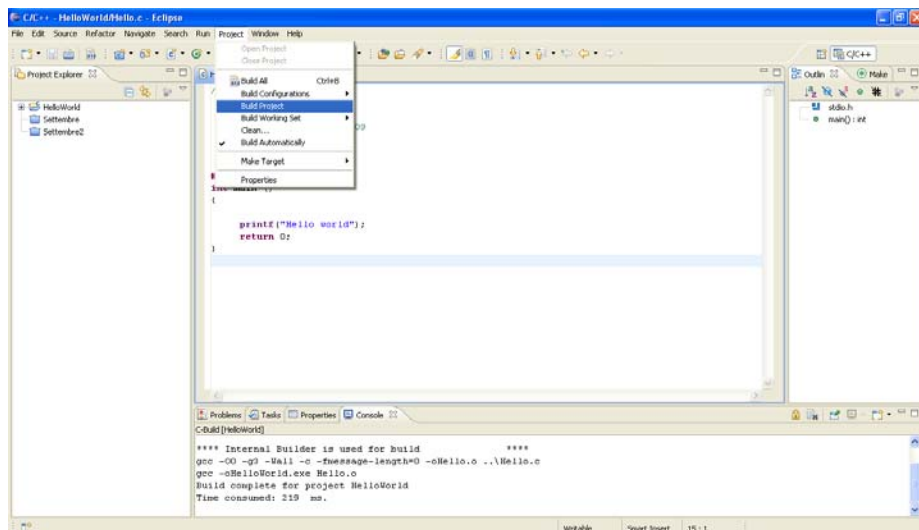
...il primo programma in C...

- L'**include** un comando che consente di richiamare ed utilizzare le librerie
 - ▢ **Stdio.h** è una libreria standard di C che consente di utilizzare il comando **printf** per la stampa a video
- La parte principale di un programma in C è il **main**
 - ▢ Il **main** è indispensabile ed unico
 - ▢ **Int** prima del main consente al programma di restituire un valore
- Le **parentesi graffe** delimitano l'inizio e la fine di una sequenza di istruzioni
- Il **punto e virgola** chiude un'istruzione
- L'istruzione **return 0;** indica che il programma è terminato con successo

...il primo programma in C

- I programmi C vengono salvati in file con estensione **.c**
- Per essere eseguiti devono essere prima compilati
- La compilazione è il processo di traduzione da linguaggio sorgente (il C nel nostro caso) a linguaggio oggetto (comprensibile alla macchina)
- Il risultato del processo di compilazione è un file eseguibile (**.exe**)

Compilare un programma...



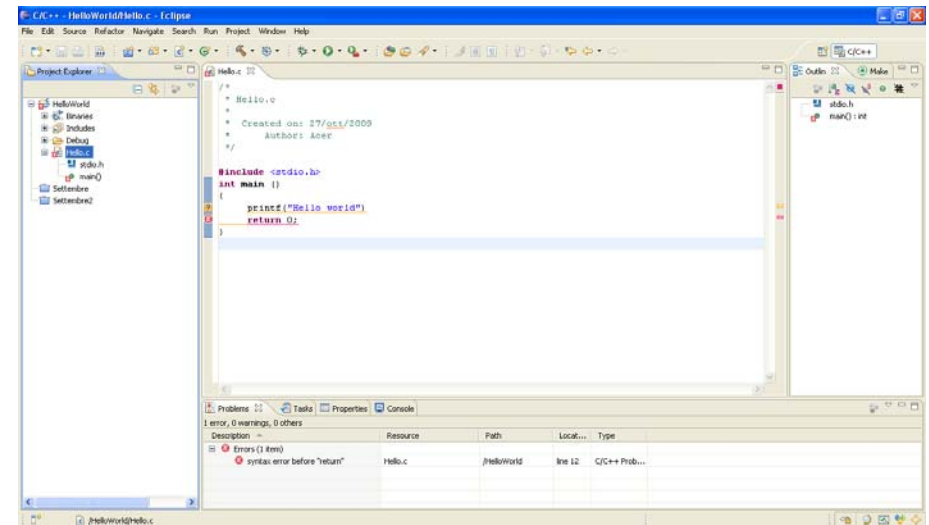
...Compilare un programma...

- Le fasi di compilazione di un programma C
 - ▢ Il codice sorgente viene controllato dal preprocessore che:
 - rimuove eventuali commenti presenti nel sorgente
 - interpreta speciali direttive per il preprocessore denotate da **"#"** (come ad esempio **#include**)
 - rileva eventuali errori sintattici
 - ▢ Il risultato del preprocessore sarà un nuovo codice sorgente "espanso" pronto per essere tradotto dal compilatore C in codice assembly

...Compilare un programma...

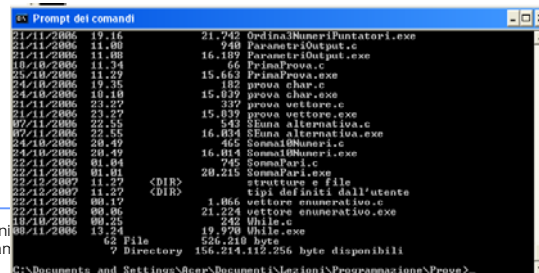
- L'assembler crea il codice oggetto salvandolo in un file oggetto (.o in Unix/Linux e .obj in Dos/Windows)
- Il Link editor ha il compito di collegare tutti i file oggetto risolvendo eventuali dipendenze e creando un unico programma eseguibile
- La compilazione individua errori di sintassi ma non può rilevare errori logici (come ad esempio un ciclo che non termina)
- In alcuni casi possono essere segnalati dei **Warning** che non costituiscono errore, ma che segnalano parti di codice strane e quindi sulle quali porre attenzione per eventuali errori logici

Errori di compilazione

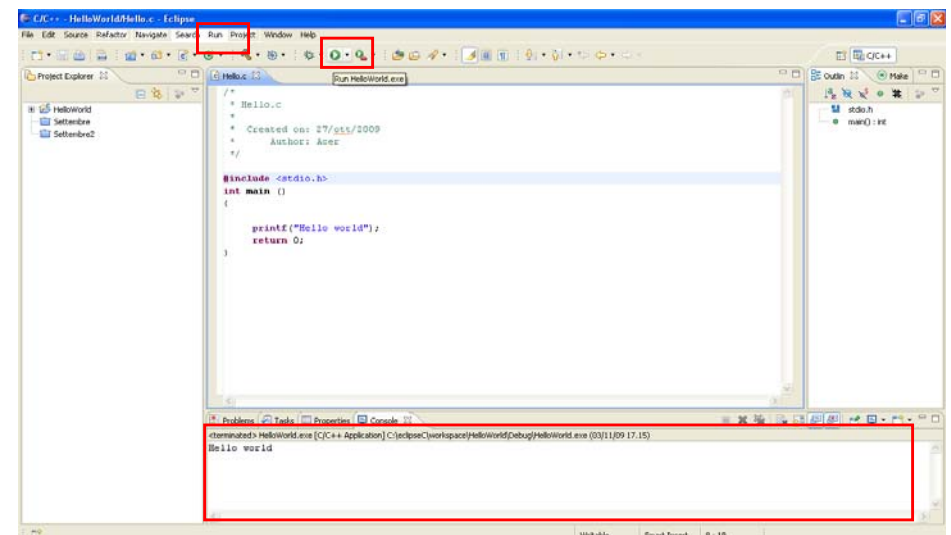


L'esecuzione di un programma...

- Un programma può essere mandato in esecuzione:
 - ▢ Direttamente dall'ambiente di programmazione cliccando sul comando Run
 - ▢ Cliccando sul file eseguibile creato in fase di compilazione
 - ▢ Dal prompt dei comandi



...L'esecuzione di un programma



Struttura di un programma C

```

Direttive per il preprocessore (es. #include)
Dichiarazione variabili globali
int main()
{
    Variabili locali al main;
    Istruzione1;
    Istruzione2;
    Istruzione3;
    ...
    IstruzioneN;
    return 0;
}

```

Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

21/43

Un secondo esempio

```

#include <stdio.h>
int main ()
{
    int a, b, media; /*definizione delle variabili
                    locali al main*/

    a=6;
    b=10;
    media=(a+b)/2;
    printf("Media tra 6 e 10: %d", media);
    return 0;
}

```

- Programma che calcola la media tra 6 e 10

Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

Una variabile è...

- ...un nome simbolico (identificatore) che denota un'area di memoria e, tramite essa, il valore contenuto
 - Contiene una rappresentazione di un oggetto su cui l'algoritmo opera
- Nel linguaggio C la definizione di una variabile è obbligatoria e deve precedere le istruzioni che la utilizzano
- La dichiarazione segue la sintassi

Tipo_variabile Identificatore_variabile

- Int a;
- Char risposta;

Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

23/43

Tipi di variabili in C...

- La definizione (o dichiarazione) del tipo di variabile è necessaria per informare l'esecutore su
 - dominio della variabile
 - insieme di operazioni effettuabili su di essa
 - modo attraverso cui ci si può riferire ad essa
- Il linguaggio C offre i seguenti tipi di dati predefiniti:
 - Int - Numero intero
 - Float - Numero reale (32 bit)
 - Double - Numero reale "lungo" (64 bit)
 - Char - Tipo Carattere

Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

24/43

Il tipo int(eger)

- È utilizzato in C per poter rappresentare i numeri interi
- [-32768, 32767] rappresenta il range di dati di tipo int
- Le operazioni possibili sono le classiche operazioni aritmetiche (+, -, *, /) e operazioni logiche di confronto
- Esempi di valori integer sono:
 - -10500 256 +36 32767

I tipi float e double (real)...

- Sono utilizzati in C per rappresentare un sottoinsieme dei numeri reali
 - alcuni numeri reali non possono essere rappresentati in nessuna notazione perché troppo grandi o troppo piccoli
- C utilizza la notazione in virgola mobile per la rappresentazione dei numeri reali
 - 2.345e2 rappresenta 234.5 (cioè $2.345 \cdot 10^2$)

...I tipi float e double (real)

- Le operazioni possibili sono le classiche operazioni aritmetiche (+, -, *, /) e operazioni logiche di confronto
- Esempi di valori float e double sono:
 - -15.0e-04 12e+5 256.36
- Esempi di valori non corretti sono:
 - 150 (manca il punto decimale) 32,56 (la virgola non è ammessa)
 - -15e-0.3 (0.3 non è un esponente valido)

Tipo Char (carattere)

- È utilizzato per rappresentare tutti i singoli caratteri alfanumerici
- Un valore di tipo char all'interno di una istruzione si racchiude tra apici
 - Iniziale = 'A'
- Le operazioni possibili sono le operazioni logiche di confronto
- Esempi di valori char sono:
 - 'A' '3' '*' ' ' ''

Gli identificatori...

- Un identificatore, dal punto di vista sintattico, è una sequenza di caratteri alfabetici e/o numerici
 - In C un identificatore DEVE iniziare con una lettera o con il simbolo `_` (underscore)
- Un identificatore **standard** è un nome che ha già un significato preciso (ad es. `printf`, `scanf`)

...Gli identificatori

- Un identificatore definito dall'utente può essere utilizzato per:
 - una variabile
 - una operazione definita dall'utente (una funzione)
- ATTENZIONE:
 - Non si possono usare le parole riservate (es. `printf`)
 - Gli identificatori **MEDIA**, **media** e **Media** sono diversi
 - Gli identificatori non possono contenere spazi

Le istruzioni di assegnazione

- Un'istruzione di assegnazione memorizza un valore o un risultato di una computazione in una variabile
- La sintassi in C:

Variabile = Espressione

- `a=6;`
- `b=secondo_valore;`
- `Media = (a+b)/2;`

dove *Espressione* può essere:

- un valore costante
- una variabile
- il risultato di un'espressione aritmetica

Le istruzioni di ingresso

- Istruzioni di ingresso consentono di acquisire informazioni dall'esterno
- L'operatore **scanf** è una funzione definita nella libreria `<stdio.h>`
- La sintassi:

scanf (formato_input, lista_input);

- `scanf ("%d", &elemento);`
- `scanf ("%c", &carattere);`

- Dove **formato_input** indica il tipo di dato che l'esecutore deve aspettarsi in input. È indicato usando un **placeholder**

Placeholder (I/O)

Placeholder

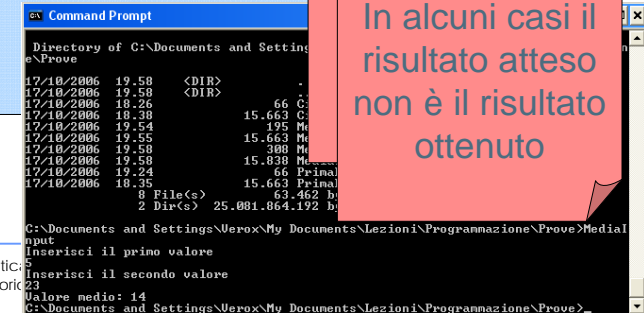
- %c
- %d
- %f
- %lf

Tipo variabile

- Char
- Int
- Float
- Double

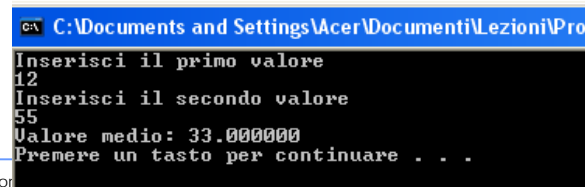
Un esempio più “generale”

```
#include <stdio.h>
int main ()
{
    int a, b, media;
    /*definizione delle variabili locali al main*/
    printf("Inserisci il primo valore \n");
    scanf("%d", &a);
    printf("Inserisci il secondo valore \n");
    scanf("%d", &b);
    media=(a+b)/2;
    printf("Valore medio: %d ", media);
    return 0;
}
```



L'esempio “generale”

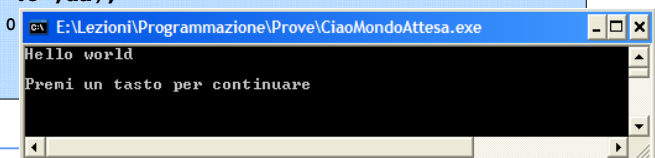
```
#include <stdio.h>
#include <stdlib.h>
int main ()
{
    int a, b; /*definizione delle variabili locali al main*/
    float media;
    printf("Inserisci il primo valore \n");
    scanf("%d", &a);
    printf("Inserisci il secondo valore \n");
    scanf("%d", &b);
    media=(a+b)/2;
    printf("Valore medio: %f \n", media);
    system ("pause");
    return 0;
}
```



Inserire una pausa nell'esecuzione...

- Per imporre delle pause nell'esecuzione è possibile
 - chiedere un input all'utente

```
#include <stdio.h>
int main ()
{
    char a;
    printf("Hello world \n\n");
    printf("Premi un tasto per continuare... ");
    scanf("%c", &a);
    return 0;
}
```

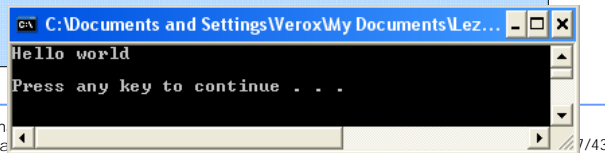


...Inserire una pausa nell'esecuzione

- Utilizzare una funzione standard di libreria
 - `system("pause")` che è incluso nella libreria `stdlib.h`

```
#include <stdio.h>
#include <stdlib.h>

int main ()
{
    printf("Hello world \n\n");
    system("pause");
    return 0;
}
```



Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

7/43

Un esercizio

- Creare l'algoritmo e poi il programma che consenta di calcolare l'indice di massa corporea (BMI) sapendo che BMI è uguale al peso (espresso in kg) diviso l'altezza (espressa in m) al quadrato

Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

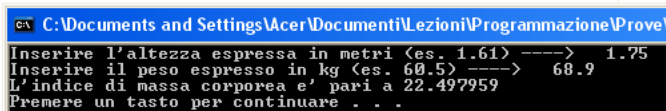
38/43

Calcolo dell'BMI

```
/*
Name: Calcola l'Indice di Massa Corporea
Copyright:
Author:
Date: 24/10/06 18.52
Description: Questo programma consente di calcolare l'indice di massa corporea (bmi)
*/
```

```
#include <stdio.h>
#include <stdlib.h>

int main ()
{
    float altezza, peso, bmi;
    printf("Inserire l'altezza espressa in metri (es. 1.61) ----> ");
    scanf("%f", &altezza);
    printf("Inserire il peso espresso in kg (es. 60.5) ----> ");
    scanf("%f", &peso);
    /* calcolo dell'indice di massa corporea */
    bmi = peso / (altezza * altezza);
    printf("L'indice di massa corporea e' pari a %f \n", bmi);
    system("pause");
    return 0;
}
```



Commenti...

- Sono parti del programma che però sono ignorate dal compilatore e non sono tradotte in linguaggio macchina.
- Sono utili perché:
 - rendono più facile la comprensione di un programma
 - per descrivere lo scopo del programma,
 - per descrivere il significato degli identificatori e/o lo scopo di ciascun passo del programma

Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

40/43

...Commenti...

- La sintassi per un commento:

```
/*
    commento
*/
```

- L'intestazione tipica di un programma inserita automaticamente con DEV-C++

```
/*
    Name:
    Copyright:
    Author:
    Date:
    Description:
*/
```

Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

41/43

Un esercizio

- Creare l'algoritmo e poi il programma che consenta di convertire una distanza espressa in miglia in una distanza espressa in km sapendo che 1 miglio (US/UK) = 1,609 km

Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

42/43

Conversione da Miglia a Km

```
/*
    Name: ConvertiInKM
    Copyright:
    Author:
    Date: 23/10/06 18.26
    Description: Questo programma converte una distanza espressa in miglia
                ad una distanza espressa in KM
*/
#include <stdio.h>
#include <stdlib.h>
#define KMS_PER_MIGLIA 1.609 /* 1609e-3 costante utile per la conversione */

int main ()
{
    float miglia, kms;
    printf("Inserire la distanza in Miglia ----> ");
    scanf("%f", &miglia);

    /* conversione della distanza in km */
    kms = KMS_PER_MIGLIA * miglia;
    printf("La distanza %f corrisponde a %f chilometri. \n", miglia, kms);
    system("pause");
    return 0;
}
```

Le costanti

- In alcuni programmi si utilizzando dati che non variano mai (o quasi mai)
- Tali dati, detti **costanti**, si definiscono nella parte del programma che contiene le dichiarative per il preprocessore con la seguente sintassi

```
#define NOME_COSTANTE valore
#define KMS_PER_MIGLIA 1.609
```

- Per convenzione si utilizzano le maiuscole per gli identificatori

Corso di laurea in Informatica e Comunicazione Digitale (sede di Taranto) – a.a. 2009-10
Laboratorio di Programmazione - Veronica Rossano

44/43