



POLITECNICO  
MILANO 1863

# INFORMATICA A – Prof. Fuggetta

Laboratorio 1

Ing. Paolo Perego, Ph.D.

Ing. Anna Maria Nestorov, MSc.

# Cos'è e a cosa serve il laboratorio di Informatica A

**Quattro** incontri con frequenza obbligatoria.

Il **quinto** incontro (19 Dicembre) sarà una prova pratica di programmazione che prevedrà la risoluzione di **1/2 esercizi in 2 ore.**

Su **Schoology** verranno caricati di volta in volta gli esercizi del laboratorio. Le soluzioni saranno pubblicate dopo qualche giorno.

**Contatti:**

[paolo.perego@polimi.it](mailto:paolo.perego@polimi.it)

[annamaria.nestorov@mail.polimi.it](mailto:annamaria.nestorov@mail.polimi.it)

# Cosa serve per programmare

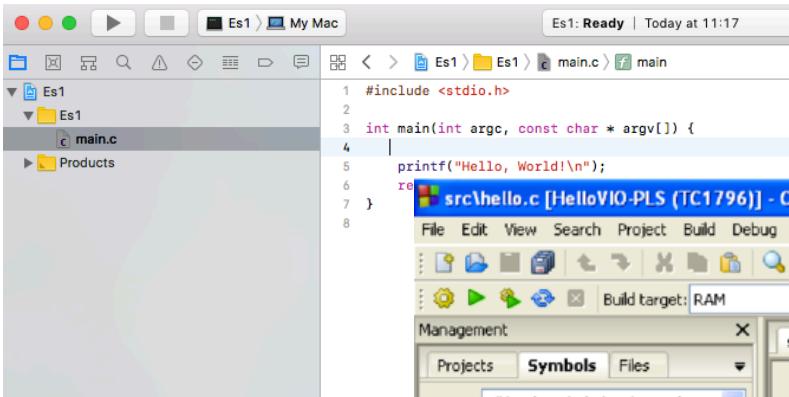
## L'IDE: Integrated Development Environment

L'IDE, detto anche ambiente di sviluppo, permette di:

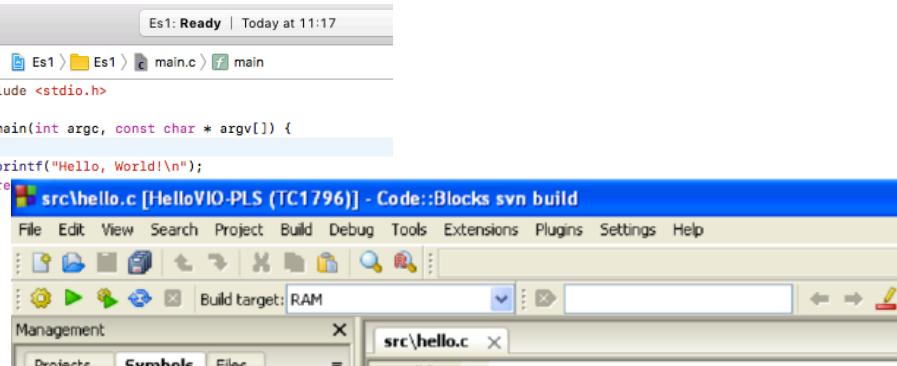
- Scrivere il codice
- Verificare eventuali errori di sintassi
- Compilare il codice
- Creare e avviare il file eseguibile

Il codice viene scritto in un file di testo con l'estensione “.c”.

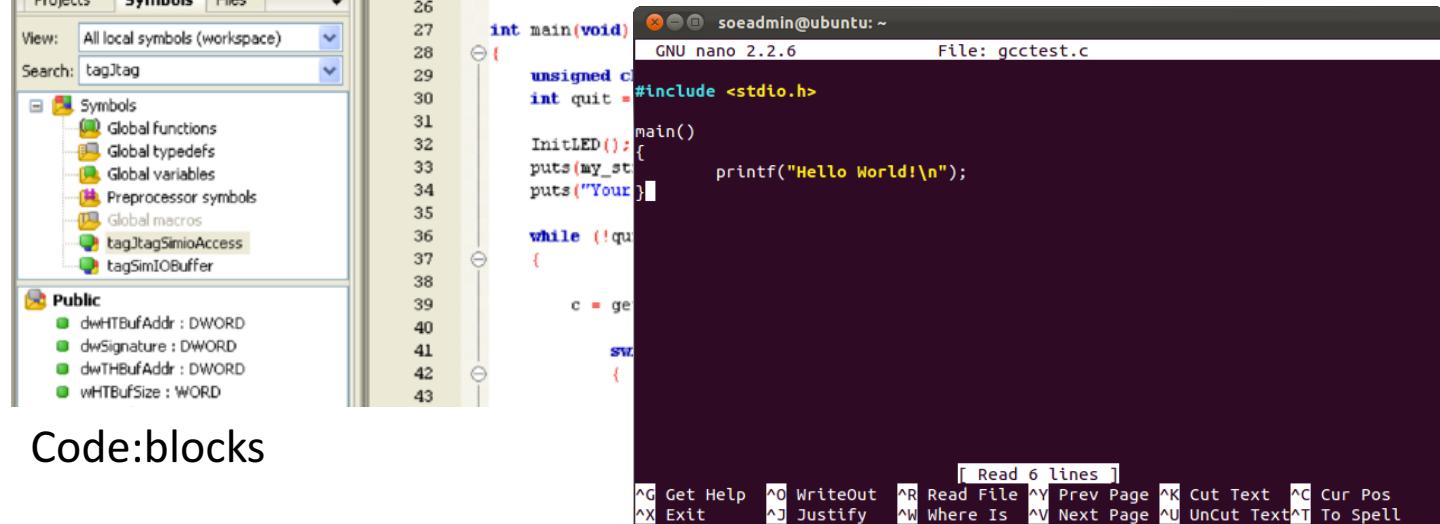
# Cosa serve per programmare L'IDE: Integrated Development Environment



Xcode x Mac



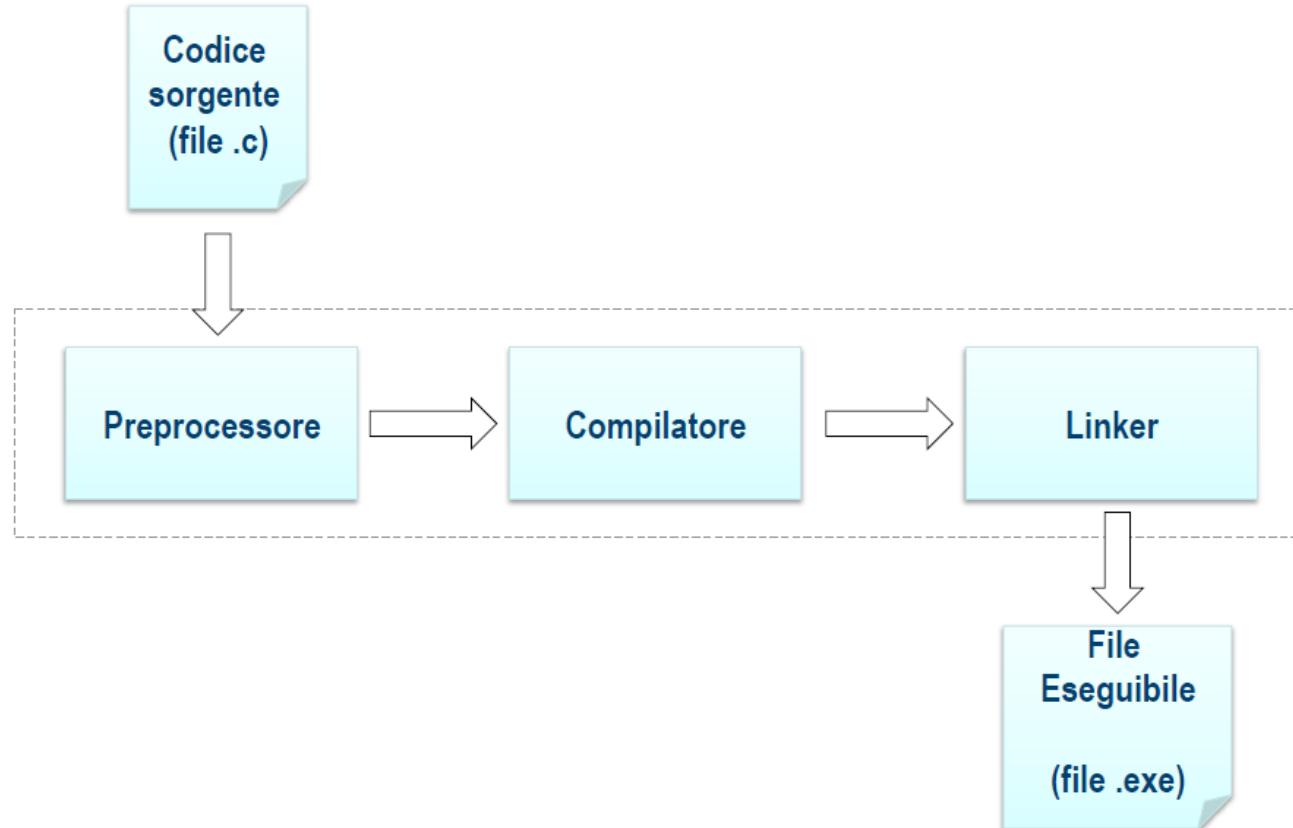
Code:blocks



Unix command line

# Cosa serve per programmare

## L'IDE: Integrated Development Environment



# Cosa utilizzare

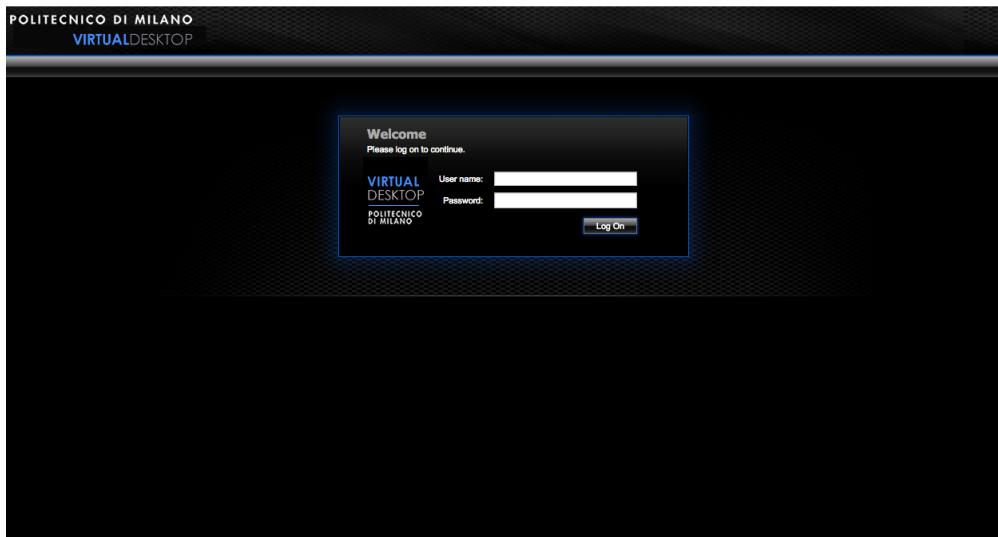
Durante le esercitazioni di laboratorio useremo **Code:Blocks**

Alternative:

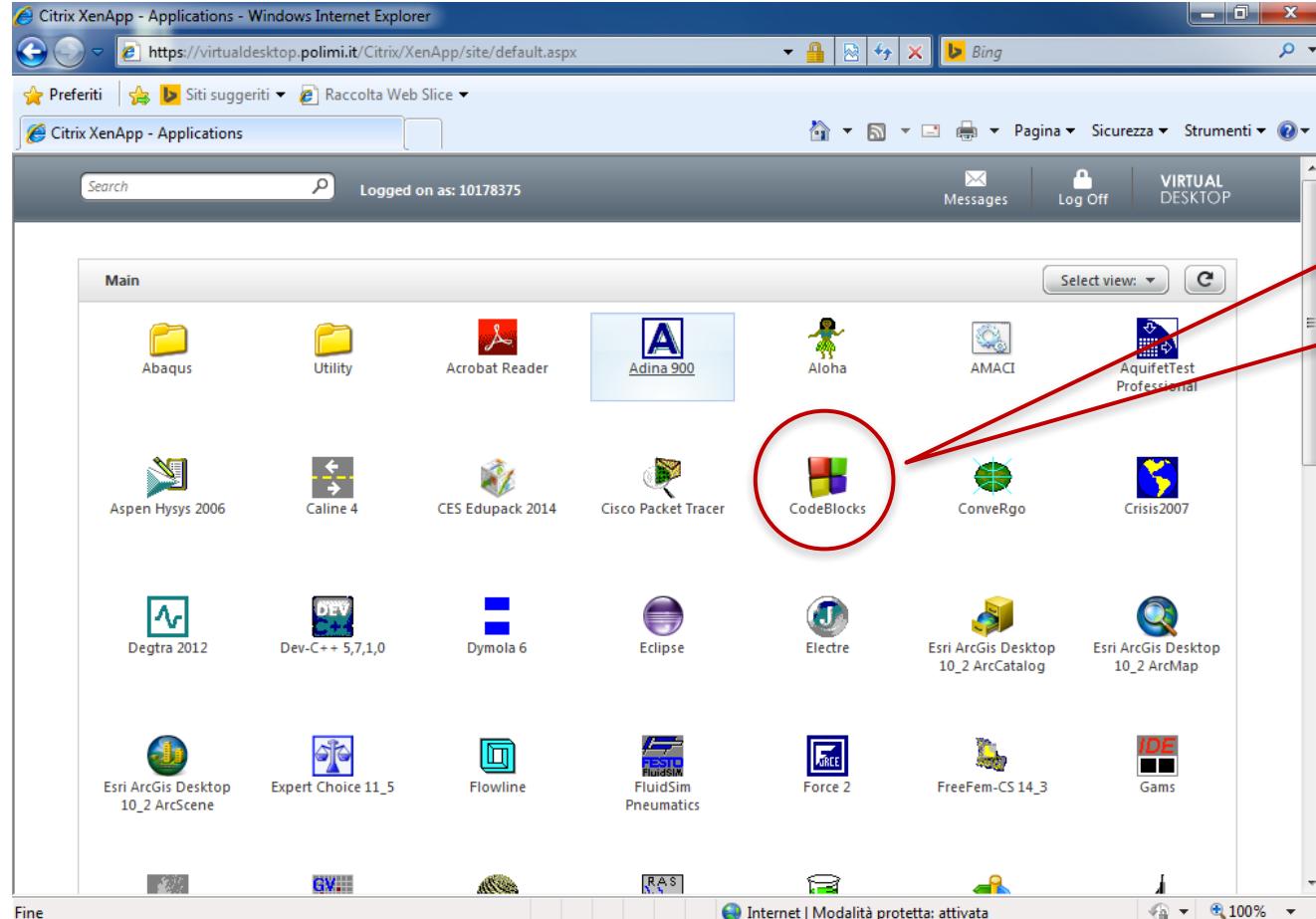
- Xcode (solo su MacOs)
- Dev-Cpp (Win Os)
- Visual Studio Code (all)
- [www.repl.it](http://www.repl.it) (da browser, anche su tablet)

# Code:Blocks sui PC del laboratorio

- Avviare Internet Explorer e andare al link:  
<https://virtualdesktop.polimi.it>
- Inserire codice persona e password

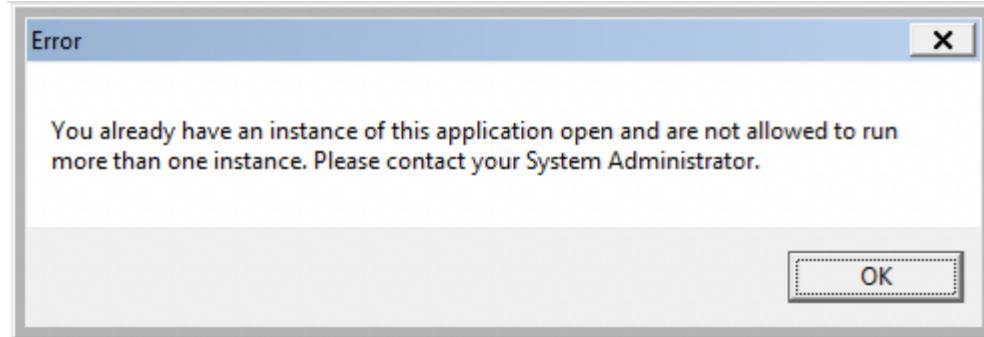


# Code:Blocks sui PC del laboratorio



Cliccare una volta sull'icona CodeBlocks ed attendere l'avvio del programma

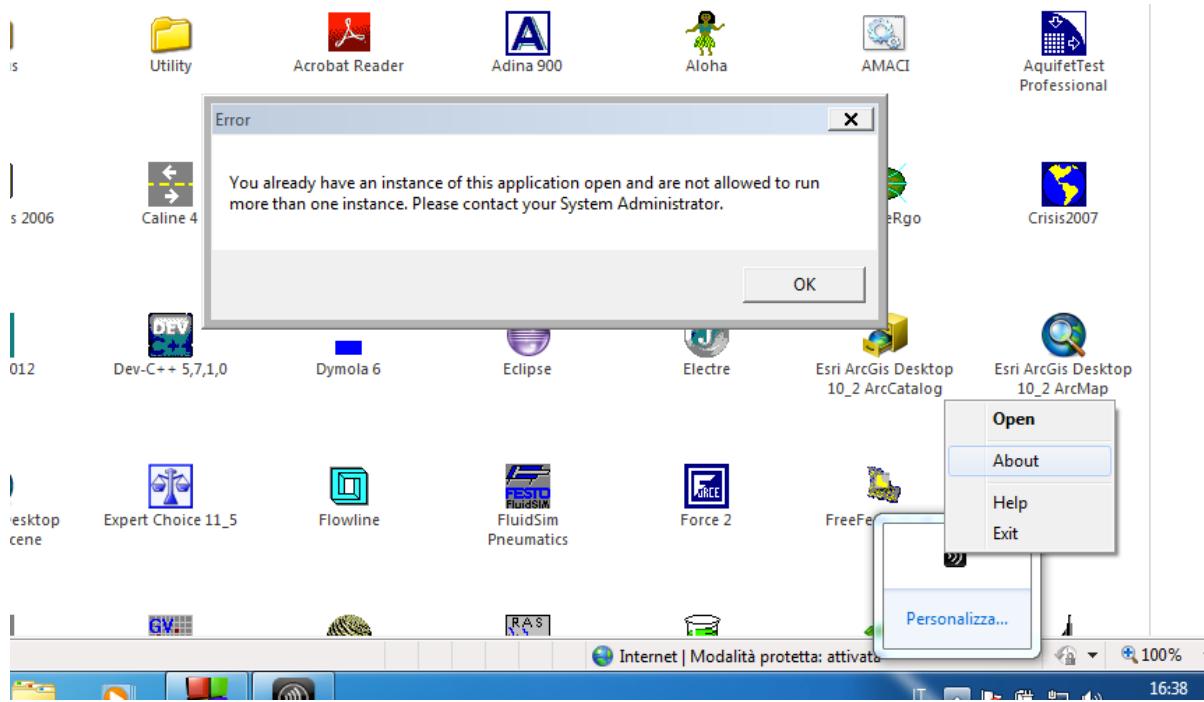
# Code:Blocks sui PC del laboratorio



Se compare questo messaggio, il programma è già in esecuzione, oppure è stato chiuso in modo inappropriato e una sua istanza è già in esecuzione

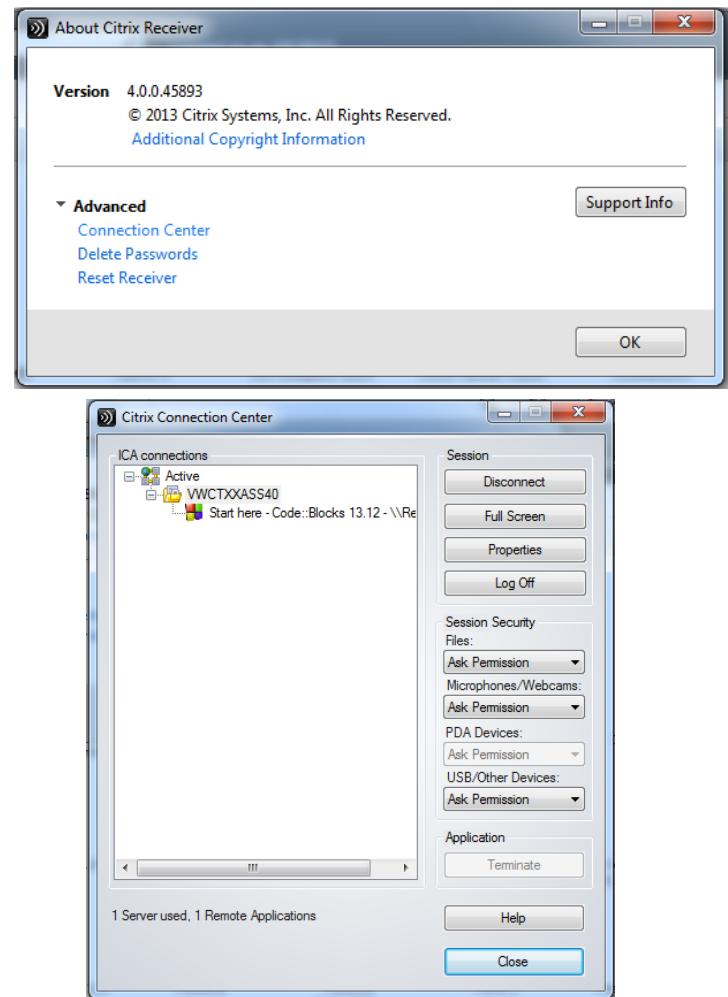
# Code:Blocks sui PC del laboratorio

Nel secondo caso, cliccare col destro sull'icona in basso a destra di **Citrix receiver** e selezionare **About**

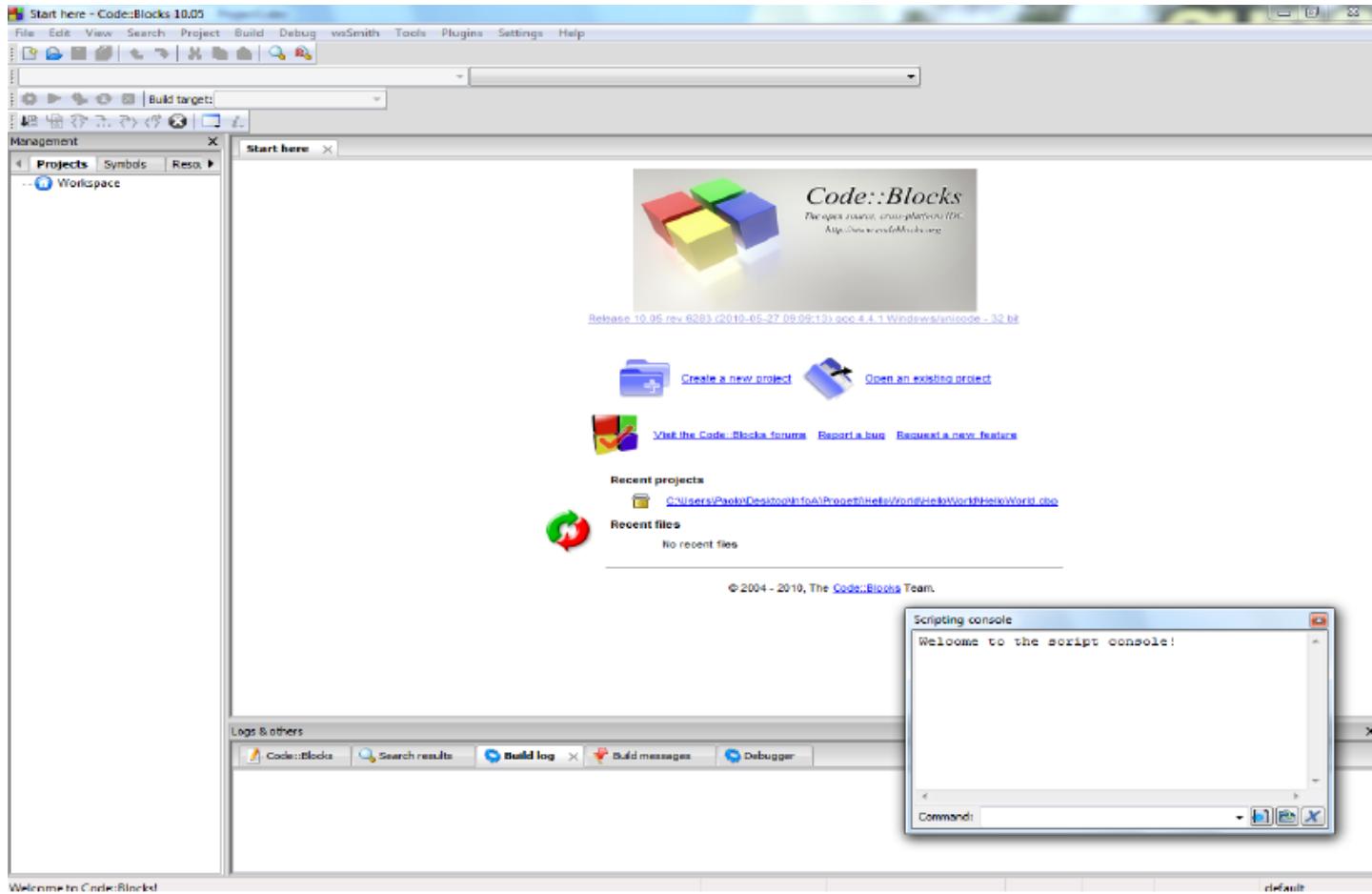


# Code:Blocks sui PC del laboratorio

- Selezionare  
**Advanced > Connection Center**
- Selezionare Codeblocks e cliccare su **Disconnect**
- **Riavviare il programma**



# Code::Blocks sui PC del laboratorio

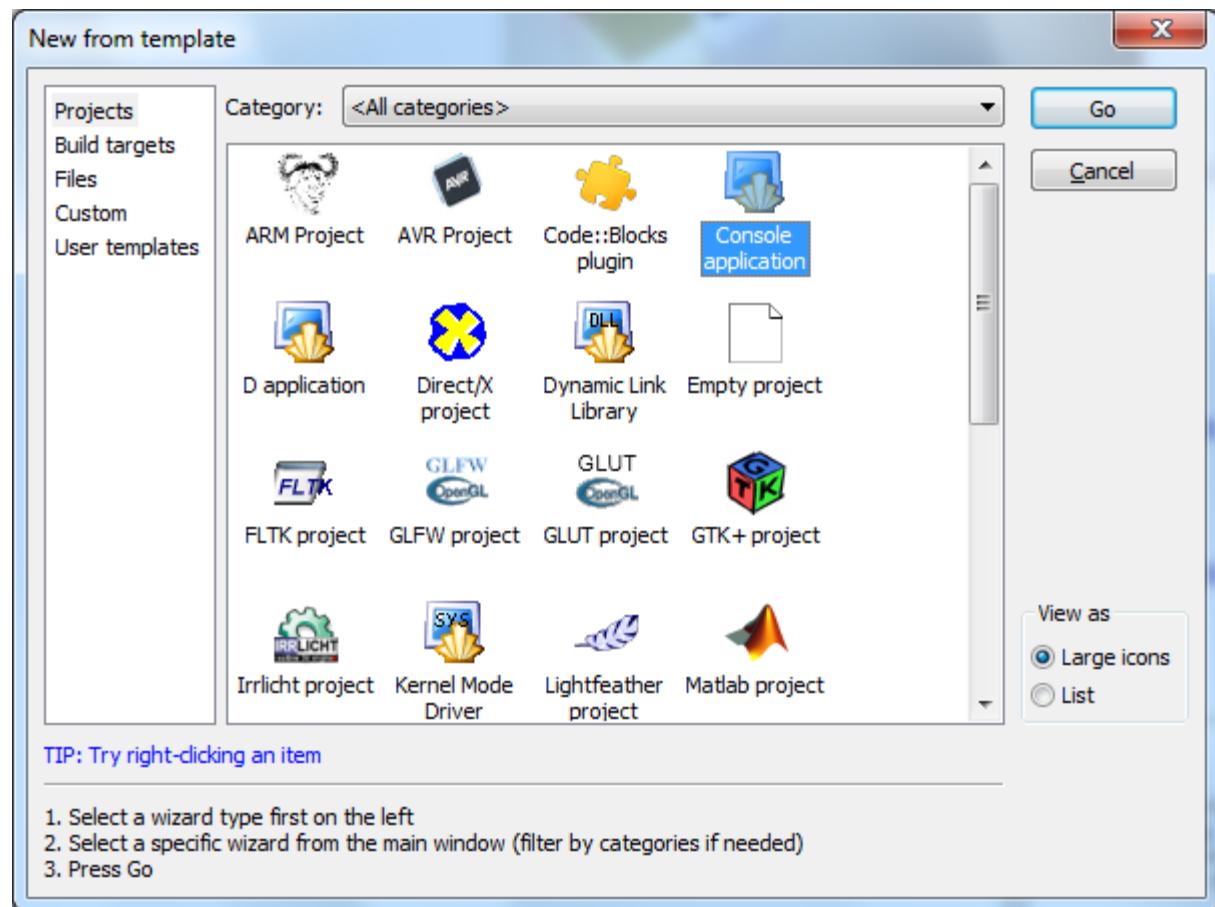


# Code::Blocks

## Creare un nuovo progetto “C”

Per creare un  
nuovo progetto:

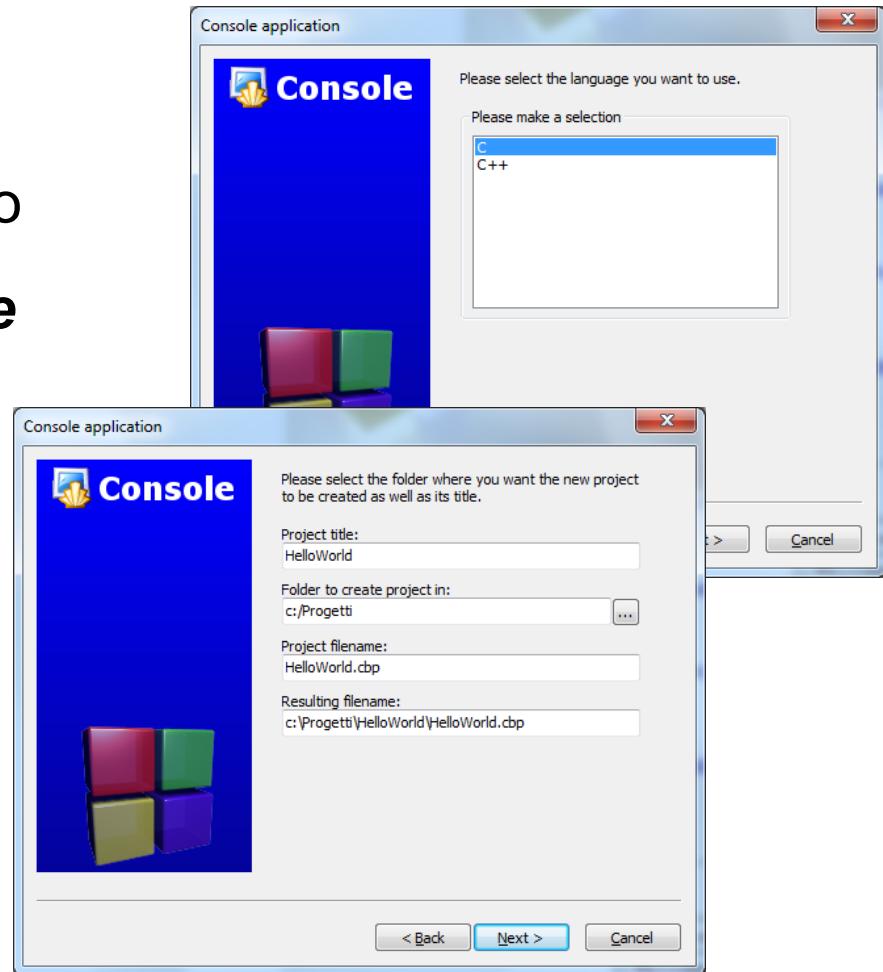
1. Menu file
2. New
3. Project
4. Select **Console application**



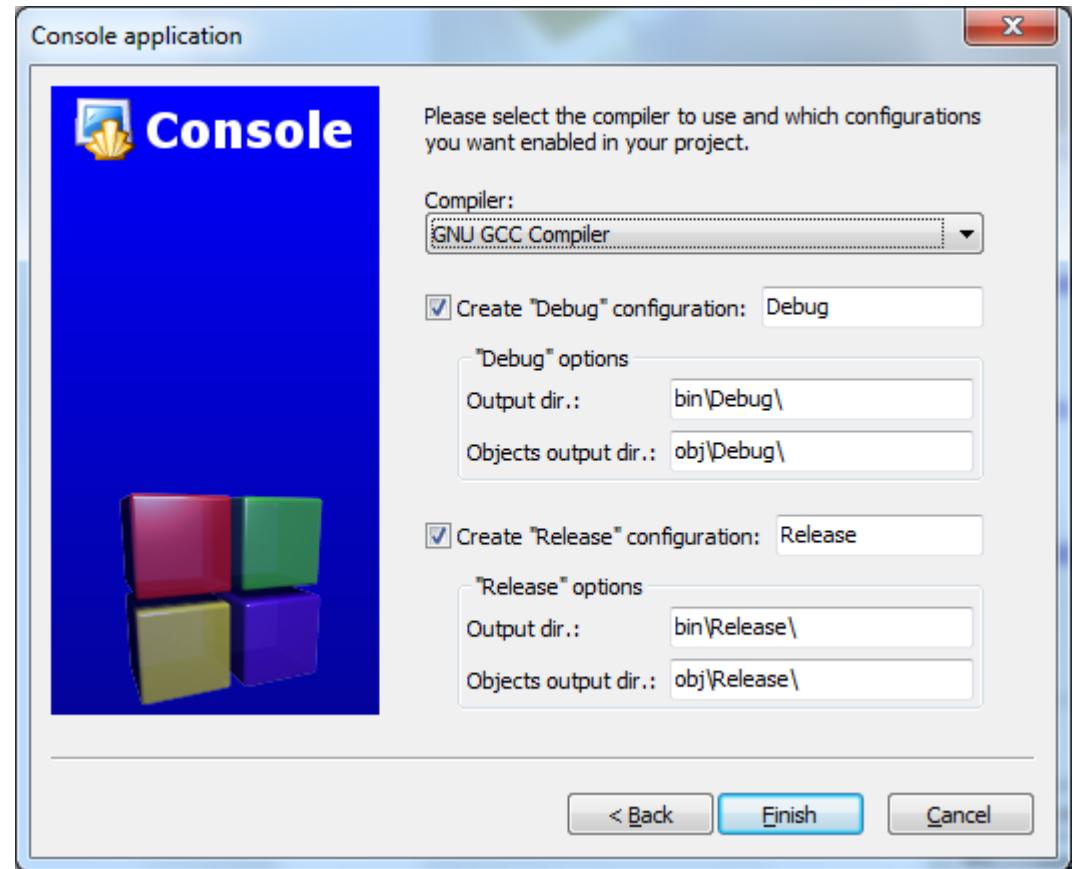
# Code:Blocks

## Creare un nuovo progetto “C”

5. Selezionare il linguaggio C
6. Selezionare il nome del progetto ed il percorso. ***Il percorso deve essere la cartella:***  
...PropriNome/Documents/

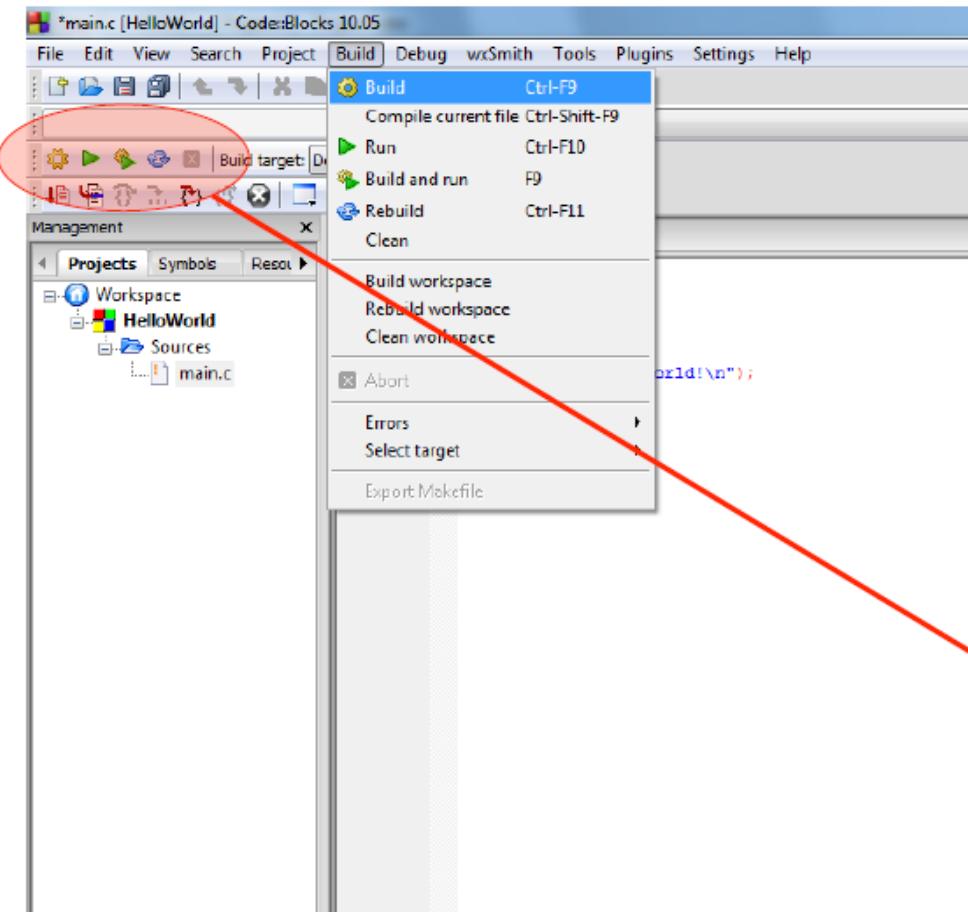


### 7. Scegliere il compilatore **GNU GCC Compiler**



# Code:Blocks

## Creare un nuovo progetto “C”



Per compilare ed eseguire il programma si può utilizzare il menu Build o le icone.

**Prima di compilare, è necessario salvare i file modificati**

**Build > Compila**

**Run > Esegue**

# Code:Blocks

## Primo programma

Proviamo a scrivere il primo programma che stampa a video il proprio nome e cognome.

```
#include <stdio.h>

int main ()
{
    println("Il mio nome è ...");
    return 0;
}
```

# Code:Blocks

## Primo programma

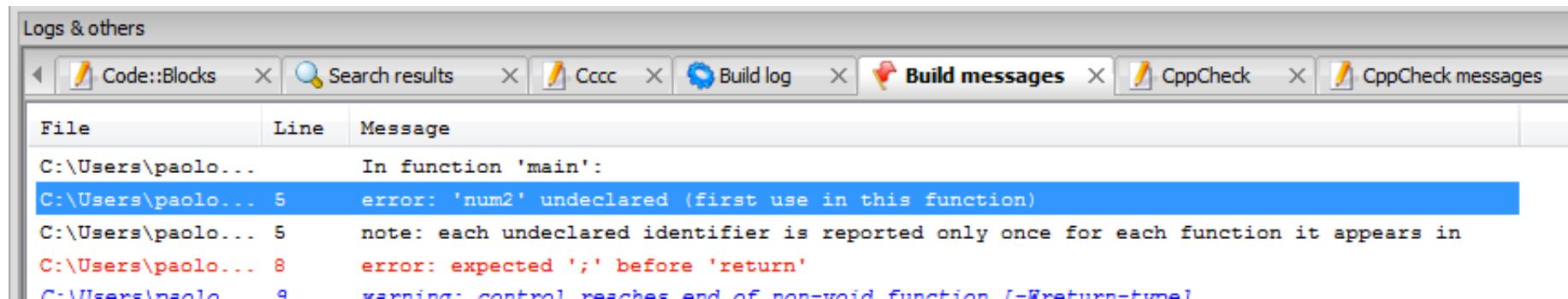
Scrivere un programma che prende in ingresso due numeri interi, calcola la loro somma e stampa a video il risultato.

```
#include <stdio.h>
int main() {
    int num1, risultato;
    scanf("%d", &num1);
    scanf("%d", &num2);
    risultato = num1 + num2;
    printf("%d", risultato)
    return 0;
}
```

*Proviamo a compilare! Cosa ci dice il compilatore?*

# Code:Blocks

## Primo programma



The screenshot shows the 'Logs & others' window of the Code:Blocks IDE. The 'Build messages' tab is active, displaying the following error output:

File	Line	Message
C:\Users\paolo...		In function 'main':
C:\Users\paolo...	5	error: 'num2' undeclared (first use in this function)
C:\Users\paolo...	5	note: each undeclared identifier is reported only once for each function it appears in
C:\Users\paolo...	8	error: expected ';' before 'return'
C:\Users\paolo...	9	warning: control reaches end of non-void function [-Wreturn-type]

- Il primo errore "num2" *undeclared* indica che la **variabile num2 non è mai stata dichiarata**
- Il secondo errore indica invece che manca un ;

Cliccando sulla riga relativa all'errore, nel codice viene indicata la riga contenente lo stesso.

# Code:Blocks

## Primo programma

Dov'è l'errore questa volta?

```
#include <stdio.h>
int main()
{
    int 1num, num2, risultato;
    scanf ("%d", &1num);
    scanf ("%d", &NUM2);
    risultato = num1 + num2;
    printf ("%d", risultato);
    return 0;
}
```

# Code:Blocks

## Primo programma

```
error: invalid suffix "num" on integer constant
```

Una variabile non può iniziare con una cifra.

Proviamo un'altra volta: dov'è l'errore ora?

```
#include <stdio.h>
int main() {
    int num1, num2, risultato;
    scanf("%d", &num1);
    scanf("%d", &num2);
    risultato = num1 + num2 ;
    printf("%d", risultato);
    return 0;
}
```

# Code:Blocks

## Primo programma

Il compilatore non da un vero e proprio errore, ma eseguendo il programma:

Windows: è possibile ricercare online una soluzione al problema.

➔ Cerca una soluzione online e chiudi il programma

➔ Chiudi il programma

➔ Esegui debug del programma



Visualizza dettagli problema

Il problema è dato dalla mancanza dell'operatore unario di indirizzo &

# Code:Blocks

## Alcuni consigli

- La maggior parte del tempo la si passa a correggere errori:  
è perfettamente normale!
- Sperimentare fa parte della programmazione
- Leggere e interpretare i messaggi del compilatore
- Commentare il codice
- Tenere manuale sotto mano
- In caso di dubbi... chiedere!

# Consegna degli esercizi

Alla prova di laboratorio verrà richiesto di consegnare il file .c contenente il programma.

Per agevolare la consegna il giorno della prova, **ad ogni laboratorio** verrà creato un *assignment* per inviare i file realizzati durante l'esercitazione. **L'invio non è obbligatorio né** verrà **valutato**; si consiglia comunque di provare l'invio in modo da abituarsi alla procedura.

*N.B.: È un esame di Informatica, la consegna corretta del file è parte integrante della prova di laboratorio. Non verranno accettati reclami per l'invio di file vuoti o errati.*

# Procedura caricamento file su Schoology

1. Effettuare il login su Schoology  
[www.schoology.com](http://www.schoology.com)
2. Selezionare il corso Informatica A: 2017-18
3. Cliccare su Laboratorio 2017/18
4. Selezionare il laboratorio del giorno

# Procedura caricamento file su Schoology

The screenshot shows the Schoology platform interface. At the top, there is a navigation bar with the Schoology logo, a search icon, and links for Home, Courses, Groups, and Resources. On the right side of the top bar, there are icons for messaging, users, and notifications, followed by the name "Paolo Perego" and a help icon.

The main content area displays a course titled "Informatica A: 2017-18". Within this course, a folder named "Laboratorio 2017/18" is visible. Below the folder, there is a section for "Laboratorio 1 – Gruppo A-L". The description states: "La consegna non è obbligatoria né sarà valutata." and "Due Monday, October 16, 2017 at 12:15 pm · Unpublished". There are "Add Materials" and "Options" buttons above this section. To the right of the section, there is a gear icon for settings.

On the left side of the screen, there is a sidebar with a profile picture of a man with a beard and a suit, labeled "Course Options". Below this, there is a "Materials" section with links to "Updates" (11), "Gradebook", "Badges", "Attendance", and "Members".

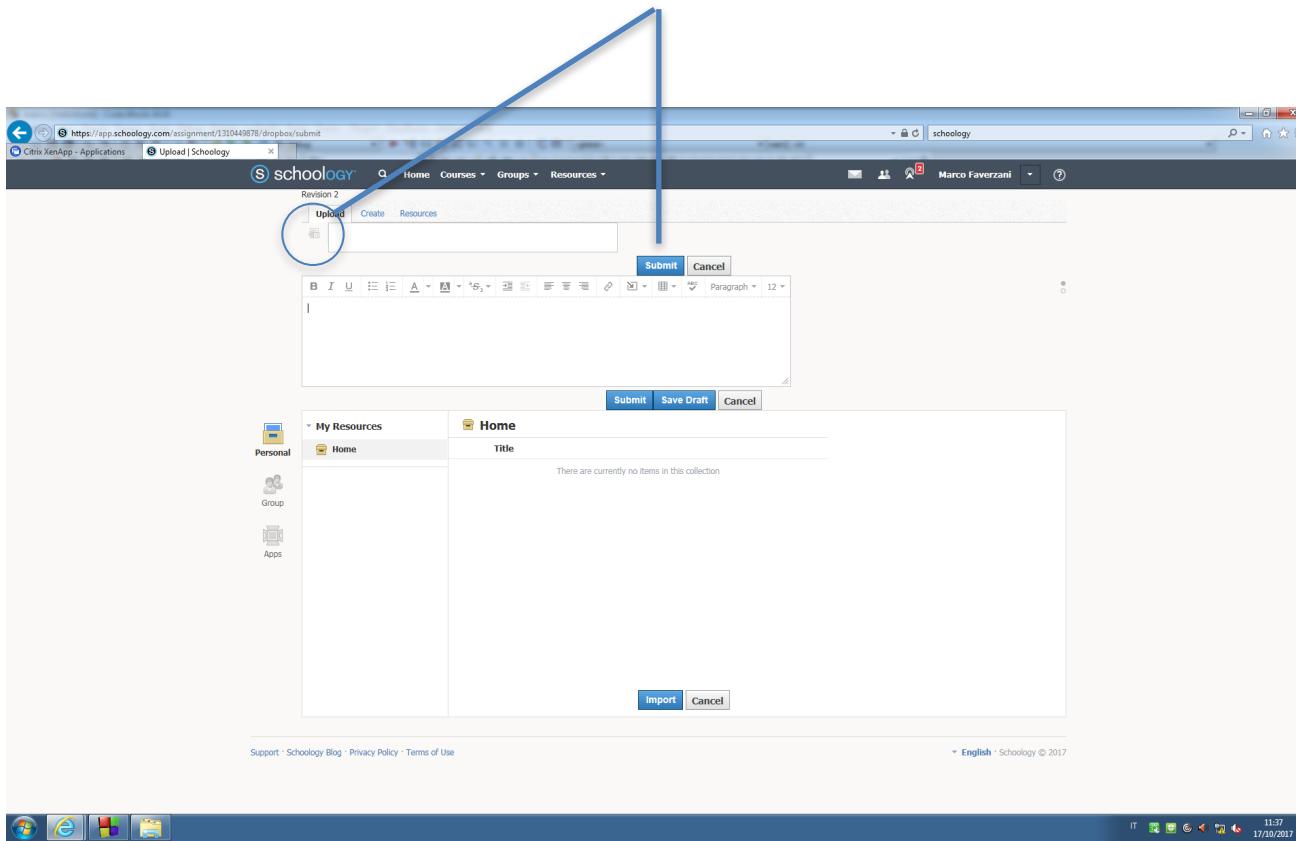
# Procedura caricamento file su Schoology

## 5. Caricare il file

The screenshot shows a Schoology assignment page for the "Apps Developmet Course: Period 2" titled "Another test". The assignment is due on Wednesday, September 26, 2012, at 11:59 pm. A red circle highlights the "Submit Assignment" button, which is located next to the "Dropbox" label.

# Procedura caricamento file su Schoology

Per caricare il file, cliccare sull'icona vicino ad Upload e successivamente il tasto submit



# Procedura caricamento file su Schoology

## Salvataggio da Code:blocks Citrix

Per poter caricare il file su Schoology occorre prima salvarlo da PC virtuale a PC locale.

L'operazione può essere effettuata in tre modi:

1. Copiare manualmente il contenuto del file .c in un file nuovo creato sul computer locale.
2. Utilizzare il comando "Save file as..." e salvarlo sul disco locale.

ATTENZIONE: Questa operazione va effettuata solamente alla fine. Una volta eseguito il comando, Codeblocks non riuscirà più a trovare il file e la compilazione darà errore.

# Procedura caricamento file su Schoology

## Salvataggio da Code:blocks Citrix

3. Utilizzare esplora risorse del computer virtuale:

- Nel browser, cliccare su **Utility** e poi **Esplora risorse**
- Cliccare su computer
- Selezionare la cartella con il **proprio nome**
- Selezionare **Documents**
- **Copiare** i file desiderati ed incollarli nella proprio **USBKEY** o in:  
**C:/User/Public/Public documents/**

**Utilizzando uno dei precedenti metodi, il file sarà accessibile anche da browser e quindi caricabile su schoology.**

# Esercizio 1

Partendo dall'esercizio somma:

- Stesso esercizio senza variabile risultato
- Stampare risultato solo se positivo, altrimenti stampare un messaggio di errore
- Sommare i due numeri solo se di segno opposto e diversi da zero
- Utilizzare un *ciclo while* per controllare che entrambi i numeri siano positivi, altrimenti richiederne l'inserimento.

## Esercizio 2

Scrivere un programma che, data una temperatura in Celsius (float o double) inserito dall'utente, la converta in Fahrenheit ed in Kelvin. Se la temperatura inserita è minore dello zero assoluto (-273,15 °C), il programma segnala un errore.

Ricordiamo che:

$$\text{Fahrenheit} = (9/5) \cdot \text{Celsius} + 32$$

$$\text{Kelvin} = \text{Celsius} + 273,15$$

Esempio: Dato  $T_c = 50^\circ\text{C}$  otteniamo  $T_f = 122^\circ\text{F}$  e  $T_k = 323,15 \text{ K}$

## Esercizio 3

Siano  $a$ ,  $b$  e  $c$  tre numeri interi; ( $a$ ,  $b$ ,  $c$ ) è una terna Pitagorica se l'equazione  $a^2 + b^2 = c^2$  risulta vera.

Consideriamo due numeri interi  $m$  e  $n$  tali che  $m > n$ , allora la terna Pitagorica generata da questi due numeri sarà costituita dai seguenti numeri interi:

$$a = m^2 - n^2, \quad b = 2mn, \quad c = m^2 + n^2.$$

Scrivere un programma che legga in input due numeri interi  $m$  e  $n$ , controlli che  $m > n$  e scriva in output la corrispondente terna pitagorica. Controllare inoltre che l'equazione sia soddisfatta.

*Suggerimento: utilizzare la funzione pow() all'interno della libreria math.h.*

```
#include <stdio.h>
#include <math.h>
void main () {
    float res = pow(2, 3);
    println("La potenza 2^3 è: %.2f", res);
}
```

## Esercizio 4

Scrivere un programma che presa in ingresso una sequenza di lunghezza ignota di numeri interi positivi, a partire dal primo numero introdotto, stampa ogni volta la media di tutti i numeri introdotti. Terminare quando il numero inserito è negativo

## Esercizio 5

Scrivere un programma che, dato un numero reale D immesso da tastiera, calcoli e stampi:

1. l'area del quadrato di lato D;
2. l'area del cerchio di diametro D;
3. l'area del triangolo equilatero di lato D.

*Suggerimento: utilizzare la funzione sqrt() e il valore di PI greco M\_PI all'interno della libreria math.h.*

```
#include <stdio.h>
#include <math.h>
void main ()
{
    float res = sqrt(144);
    println("Radice quadrata di 144 è: %.2f", res);
}
```

## Esercizio 6

Scrivere un programma che prende in ingresso un numero positivo e stampa ogni volta il fattoriale del numero appena inserito. Continuare a richiedere numeri interi fino a quando il numero inserito è negativo.

## Esercizio 7

Scrivere un programma per il "ripasso delle tabelline". Il programma riceve dall'utente un numero N compreso tra 2 e 12, e stampa la tabellina di N nel modo seguente. Esempio:

Tabellina da ripassare: 4  $4 \times 1 = 4$   $4 \times 2 = 8$  ...

$4 \times 10 = 40$

Estensione: Il programma riceve due interi N e K, e stampa la tabella di tutte le tabelline dei numeri da 1 a N, limitandosi ai primi K valori.