Università di Ferrara
Dipartimento di Ingegneria

Esercitazioni di

FONDAMENTI DI INFORMATICA MODULO B

ESERCITAZIONE 1.0

Tutor
Arnaud Nguembang Fadja: ngmrnd@unife.it
Damiano Azzolini: damiano.azzolini@student.unife.it

Passare argomenti al programma

Quando si invoca un programma è possibile passare degli argomenti ad esso durante la chiamata.

Gli argomenti ricevuti saranno stringhe (array di char) quindi l'insieme di argomenti sarà contenuto in un array di array di char.

Il programma deve poter leggere questi argomenti in modo da poterli gestire.

1. Settare la funzione main in modo tale da considerare gli argomenti passati. Per fare ciò il main deve prendere in ingresso due parametri nel seguente ordine:

- Un numero intero (n di argomenti + 1) -> **`int argc`**
- Un array di stringhe -> **`char *argv[]`**

Esercizio 1. Passaggio di argomenti

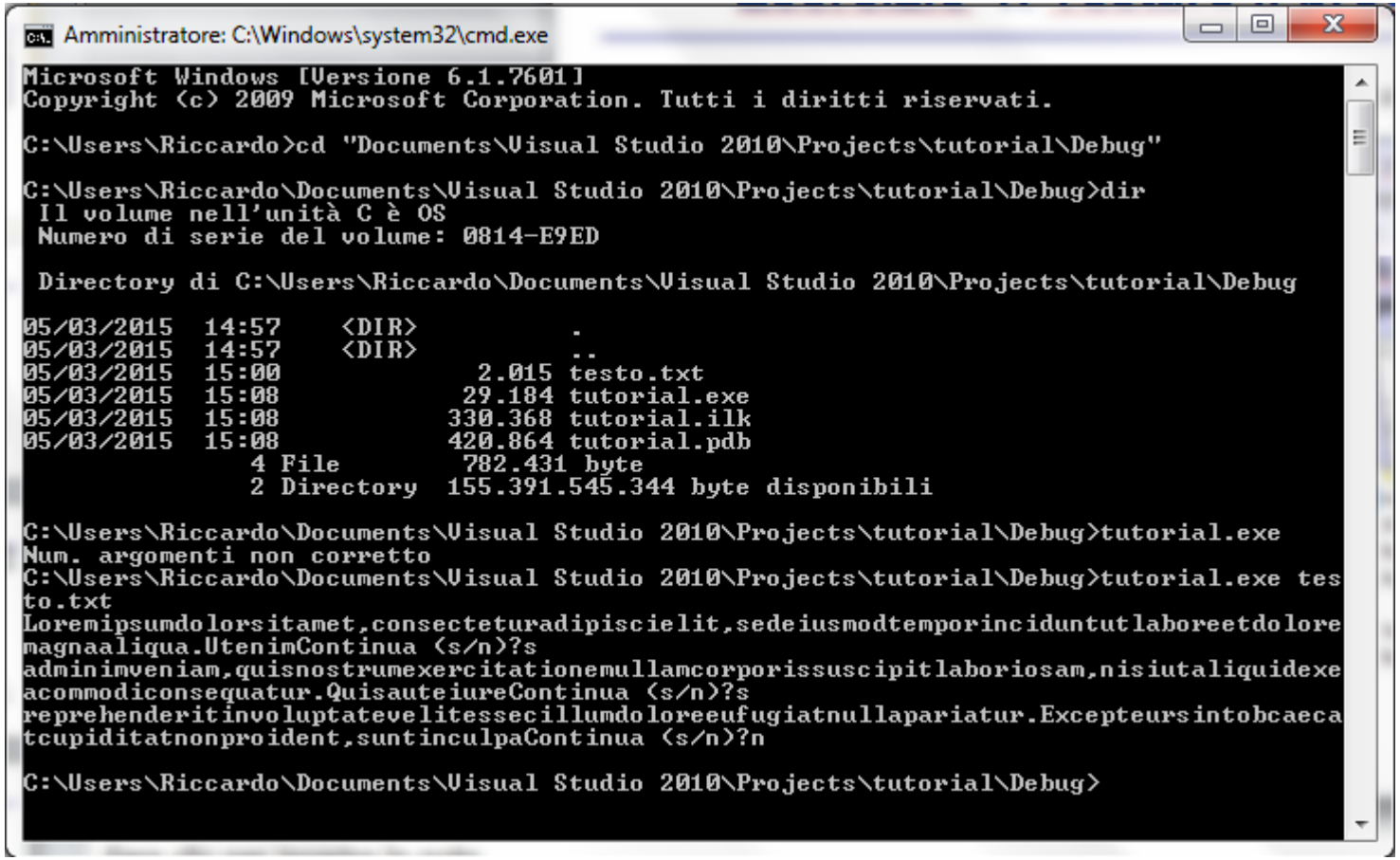
Realizzare un programma C che dato come argomento il nome di un file (di testo), ne visualizzi il contenuto a pagine (20 righe alla volta). Ad ogni visualizzazione chiedere all'utente se vuole visualizzare altre righe o fermarsi.
Di seguito una traccia per iniziare:

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

main(int argc, char *argv[]) {

      // codice

}
```

Esercizio 1. Testare il programma

Per testare il programma bisogna passare il nome del file come argomento. Abbiamo due possibilità:

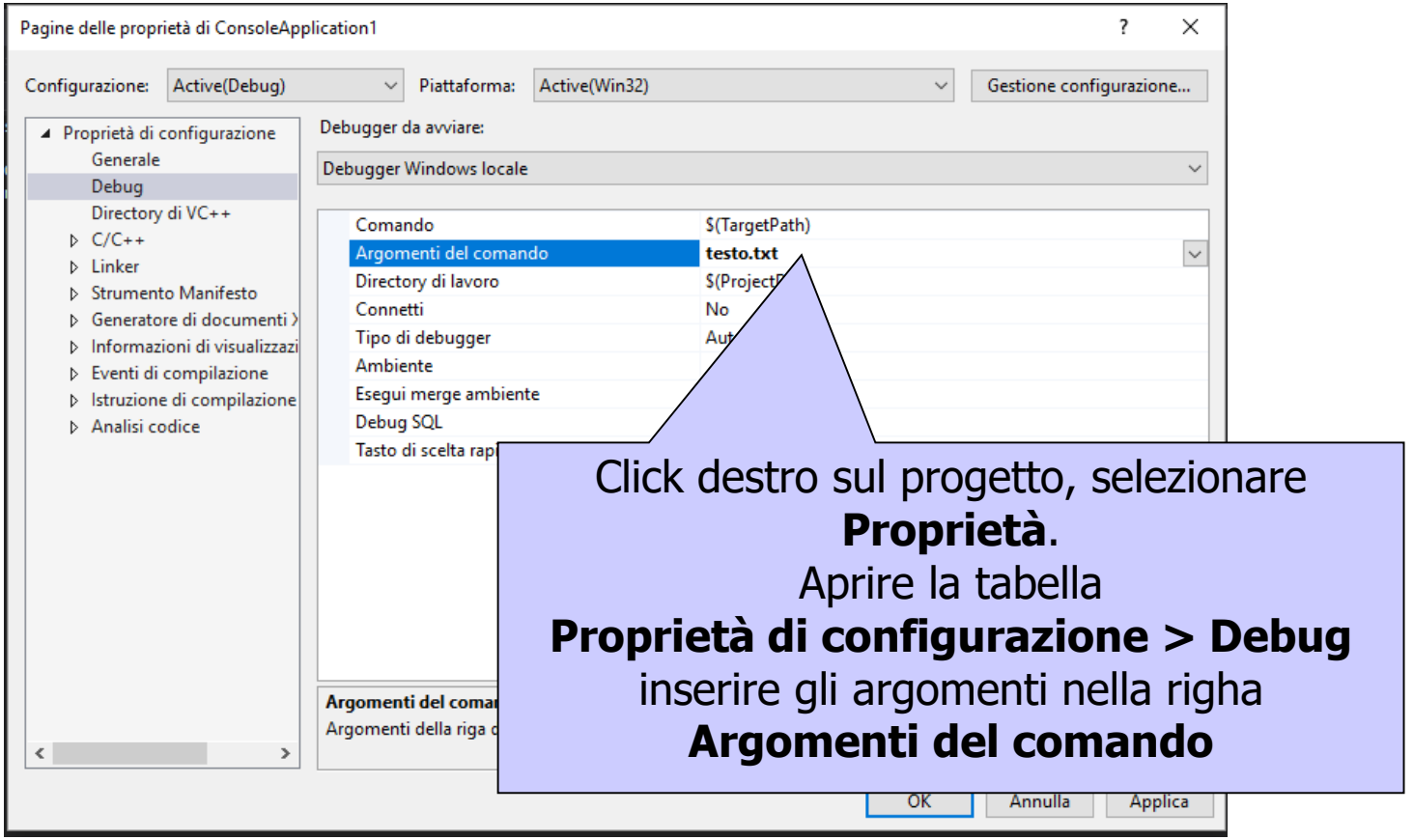
1. Eseguire da terminale il programma



Esercizio 1. Testare il programma

Per testare il programma bisogna passare il nome del file come argomento. Abbiamo due possibilità:

2. Settare l'argomento in Visual Studio



Esercizio 2. Funzioni come parametri

Si realizzi un programma che calcola l'integrale di una funzione **f : R -> R** in un intervallo dato [a,b].

Definire una funzione che restituisca il valore dell'integrale, con la seguente interfaccia

```
/*
 * func funzione integranda
 * a,b estremi intervallo, N numero sotto-intervalli
 */
double rettangoli(double (*func)(double), double a, double
b, double N);
```

- Il main legge da input **a**, **b**, **N**; poi chiama **rettangoli** e calcola l'integrale di funzioni reali.
- Definire alcune funzioni da passare a **rettangoli**, per esempio una funzione per calcolare il quadrato e una per calcolare il cubo di un numero reale
- Strutturare il programma su più file

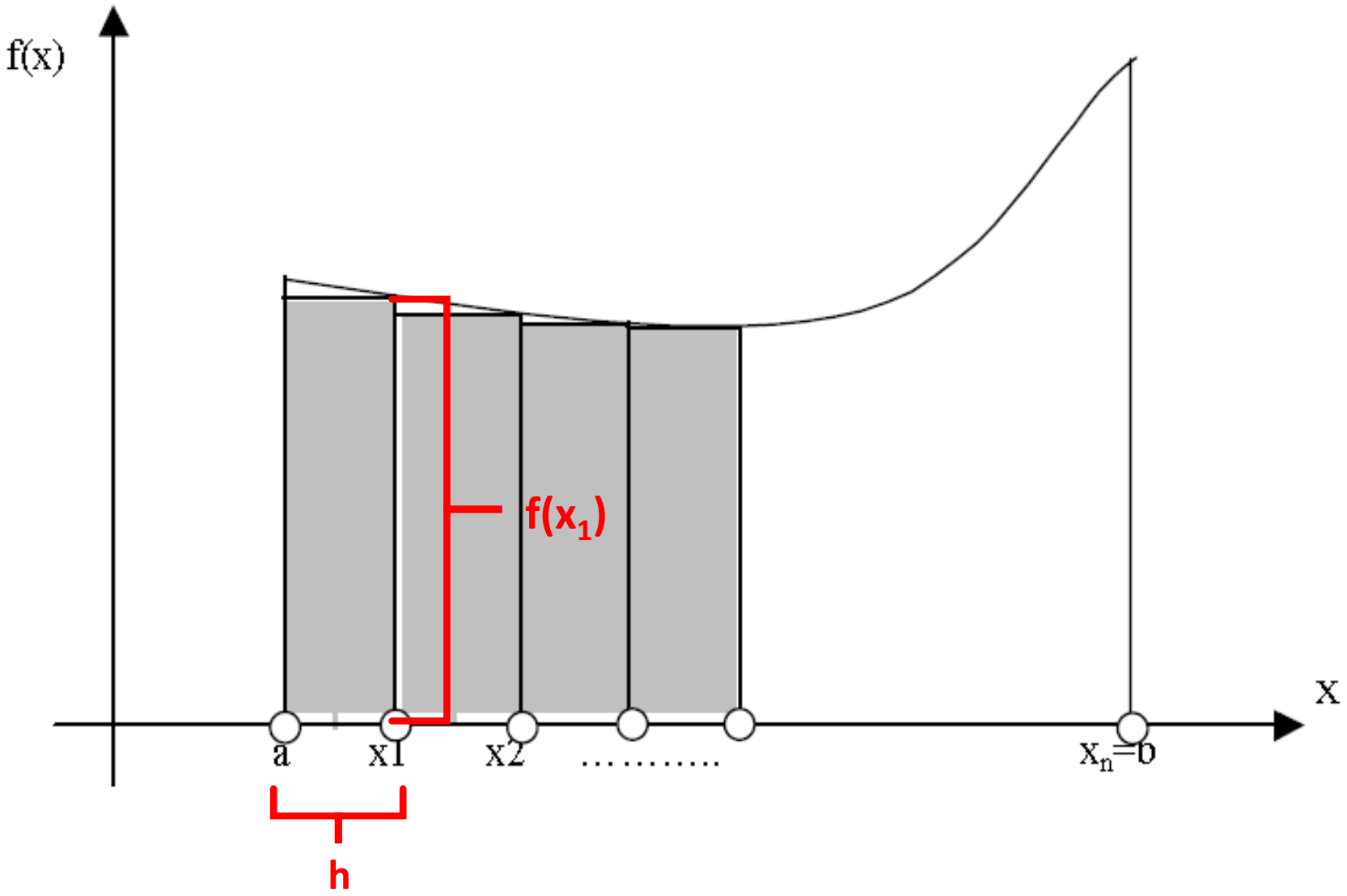
Esercizio 2. Nota

Viene indicato di seguito come specificare il parametro formale di tipo funzione nella dichiarazione della funzione "**`rettangoli`**":

```
double (*func) (double)
   ||      ||      ||
   \/      \/      \/
  (1)     (2)     (3)
```

(1) ="**tipo** del parametro di ritorno della funzione func";
(2) = "**nome** formale della funzione referenziata (puntatore)";
(3) ="**lista dei tipi** di parametri di ingresso della funzione func";

Esercizio 2. Metodo dei rettangoli



Esercizio 3. Variabili globali

Si realizzi un ***componente dotato di stato interno*** che rappresenta un ***contatore***, con operazioni:

- **`void reset( void )`**
- **`void inc( int k )`**
- **`int getValue( void )`**

Il main, che utilizza il contatore, deve in successione:

- resettarlo;
- acquisire da input una sequenza di interi positivi (terminata da 0) e usarli come incremento del contatore.
- stampare lo stato finale
- *La variabile contatore deve essere **globale**, locale al file contenente le funzioni del contatore.*
- Strutturare il programma su più file

Esercizio 4. Variabili statiche

Si realizzi un ***componente dotato di stato interno*** che rappresenta un ***contatore*** (variabile globale con scope limitato), con operazioni:

- **`void reset( void )`**
- **`void inc( int k )`**
- **`int getValue( void )`**

Il main, che utilizza il contatore, deve in successione:

- resettarlo;
- acquisire da input una sequenza di interi positivi (terminata da 0) e usarli come incremento del contatore.
- stampare lo stato finale
- *La variabile contatore deve essere **statica**, locale al file contenente le funzioni del contatore.*
- Strutturare il programma su più file

Esercizio 4. Variabili statiche

Una volta concluso si può anche…

- Verificare che lo *scope* della variabile sia solo il file contatore.c (“usare” la variabile nel main -> errore di compilazione: variabile non definita)