

# Corso di Fondamenti di Informatica



## Lezione 9

**Nicola Capuano**

Dipartimento di Scienze Aziendali, Management  
& Innovation Systems

[ncapuano@unisa.it](mailto:ncapuano@unisa.it)

## Nuovo Orario

Date	Aule	Orari
Lunedì 27 Novembre	Laboratorio	14.30-16.30
Mercoledì 29 Novembre	Aula B	12.30-14.30
Giovedì 30 Novembre	Laboratorio	8.30-10.30
Giovedì 30 Novembre	Laboratorio	12.30-14.30
Lunedì 4 Dicembre	Laboratorio	14.30-16.30
Mercoledì 6 Dicembre	Aula 6	14.30-16.30
Giovedì 7 Dicembre	Laboratorio	8.30-10.30
Giovedì 7 Dicembre	Laboratorio	12.30-14.30

# Programma del Corso

## Modulo 2 – Laboratorio di tecnologie dell'informazione e della comunicazione (aggiornamento)

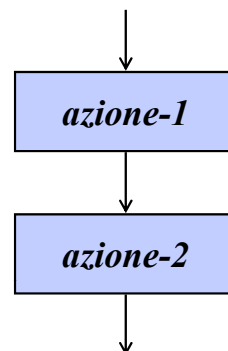
- Algoritmi e Flow Chart
- Sistemi informativi e DBMS

## Flow Chart (Approfondimenti)

Un flow chart può essere realizzato collegando i blocchi tramite **tre schemi differenti**

### Sequenza

Le operazioni vengono eseguite l'una dopo l'altra



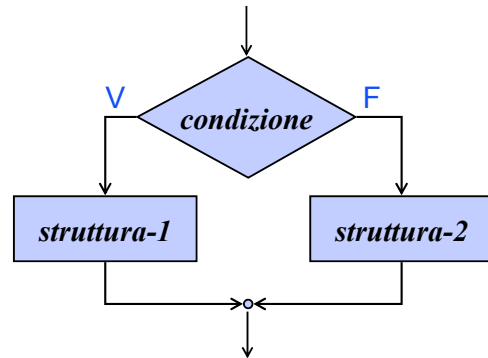
## Flow Chart (Approfondimenti)

Un flow chart può essere realizzato collegando i blocchi tramite **tre schemi differenti**

### Selezione

l'algoritmo decide quale strada seguire in base al verificarsi di una condizione

In AlgoBuild:  If



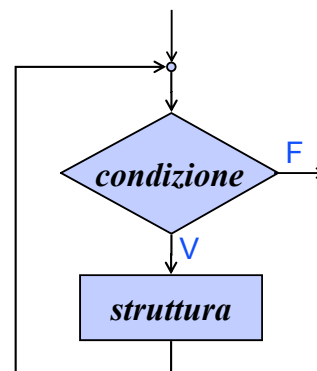
## Flow Chart (Approfondimenti)

Un flow chart può essere realizzato collegando i blocchi tramite **tre schemi differenti**

### Iterazione (while)

Una sequenza di istruzioni viene ripetuta più volte finché una condizione non risulta falsa

In AlgoBuild:  While



La condizione viene verificata **prima di eseguire le istruzioni** dell'iterazione (che possono anche non essere eseguite)

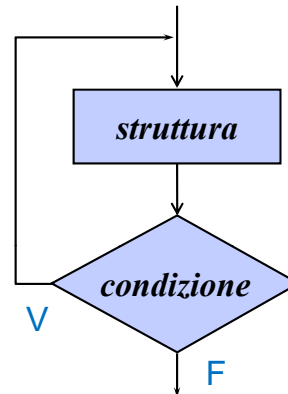
## Flow Chart (Approfondimenti)

Un flow chart può essere realizzato collegando i blocchi tramite **tre schemi differenti**

### Iterazione (do-while)

Una sequenza di istruzioni viene ripetuta più volte finché una condizione non risulta falsa

In AlgoBuild:  Do-While



La condizione viene verificata **dopo aver eseguito le istruzioni** dell'iterazione (che vengono eseguite sempre almeno una volta)

## Flow Chart (Esempi)

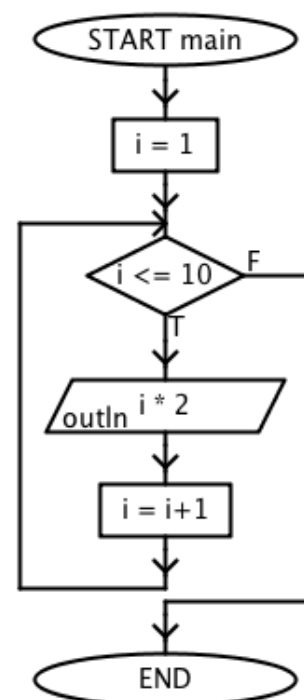
Descrivere, mediante diagramma di flusso, un algoritmo che stampa i **primi 10 numeri pari**

### Iterazione di tipo **while**

- La condizione viene verificata prima di eseguire le istruzioni

### La variabile **i** è detta **contatore**

- Ha la funzione di contare **il numero di volte** che si è eseguita l'iterazione



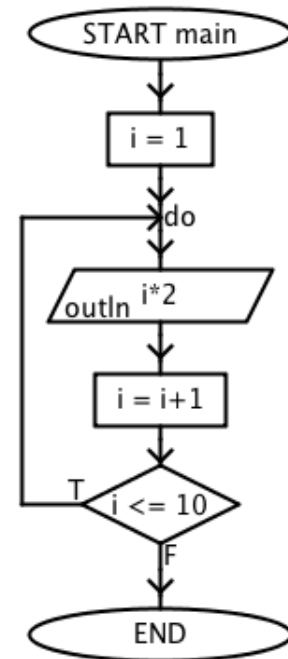
## Flow Chart (Esempi)

Descrivere, mediante diagramma di flusso, un algoritmo che stampa i **primi 10 numeri pari**

Versione alternativa:

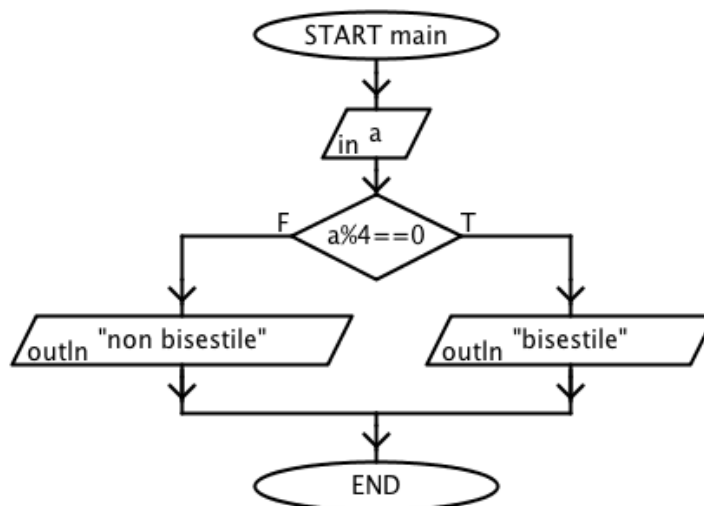
Iterazione di tipo **do-while**

- La condizione viene verificata dopo aver eseguito le istruzioni



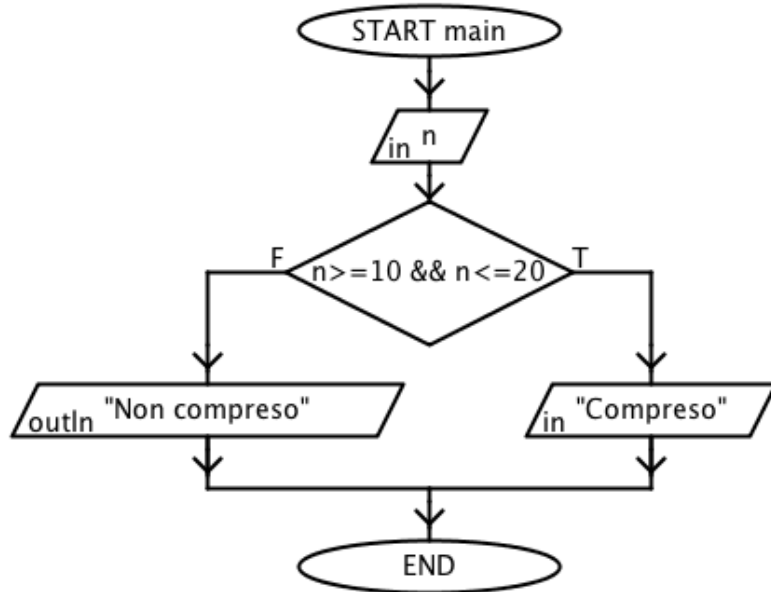
## Flow Chart (Esercizi)

Descrivere, mediante diagramma di flusso, un algoritmo che verifica se un anno **a** richiesto in input è bisestile



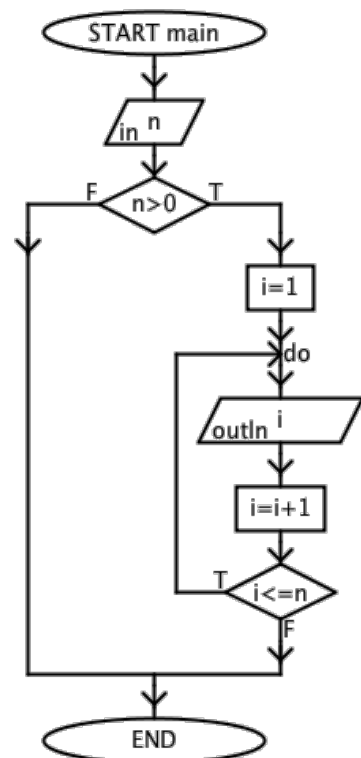
## Flow Chart (Esercizi)

Descrivere, mediante diagramma di flusso, un algoritmo che verifica se un numero **n** richiesto in input appartiene all'intervallo [10,20]



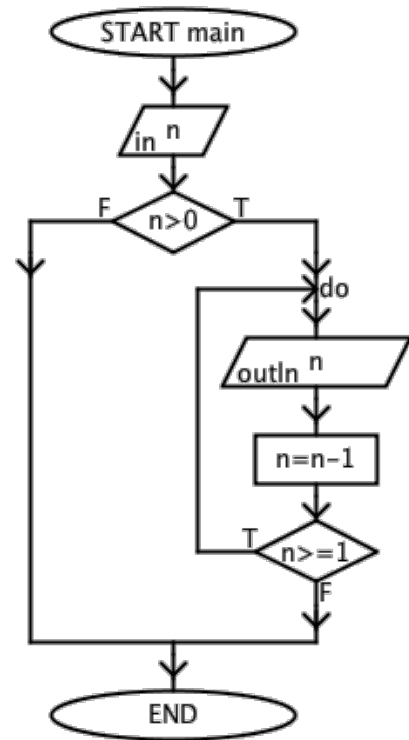
## Flow Chart (Esercizi)

Descrivere, mediante diagramma di flusso, un algoritmo che, dato un intero **n** richiesto in input, dopo aver verificato che **n** è positivo, visualizza in sequenza i numeri da 1 fino a **n**



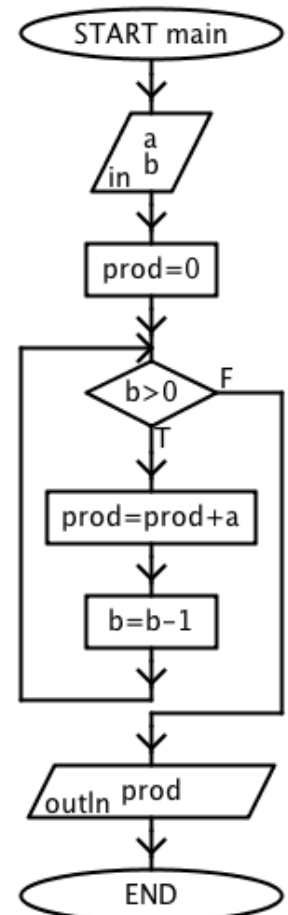
## Flow Chart (Esercizi)

Descrivere, mediante diagramma di flusso, un algoritmo che, dato un intero **n** richiesto in input, dopo aver verificato che **n** è positivo, **visualizza in sequenza i numeri da n fino a 1**



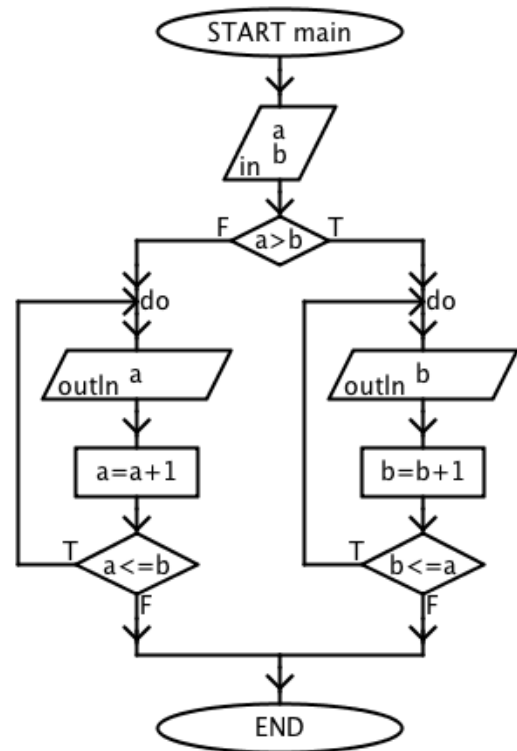
## Flow Chart (Esercizi)

Descrivere, mediante diagramma di flusso, un algoritmo che richiede in input due interi **a** ed **b** e ne calcola il **prodotto mediante il metodo delle somme ripetute**



## Flow Chart (Esercizi)

Descrivere, mediante diagramma di flusso, un algoritmo che, dati due interi in input **a** e **b** visualizzi **tutti gli interi compresi tra il minimo ed il massimo** dei due



## Flow Chart (Esercizi)

Descrivere, mediante diagramma di flusso, un algoritmo che calcola la **somma di 10 numeri** richiesti in input

