

Informatica B - Esercitazione 1

Algoritmi e schemi a blocchi

Stefano Cereda

stefano.cereda@polimi.it

25/09/2018

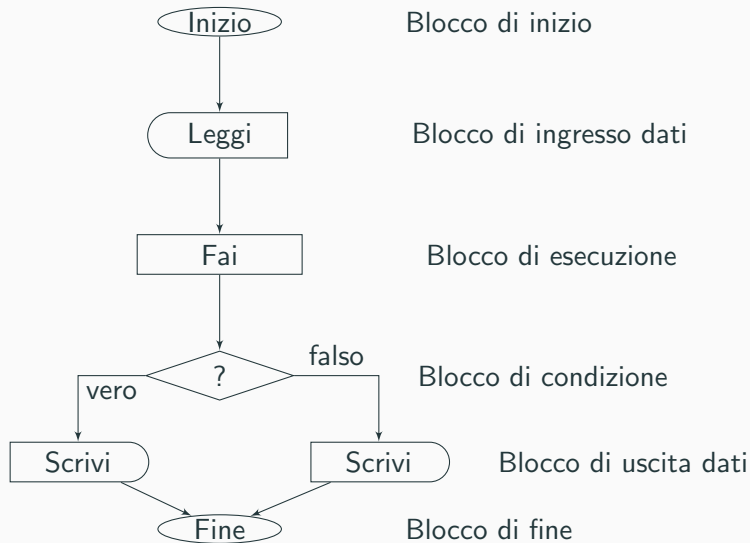
Politecnico di Milano



Approccio ai problemi

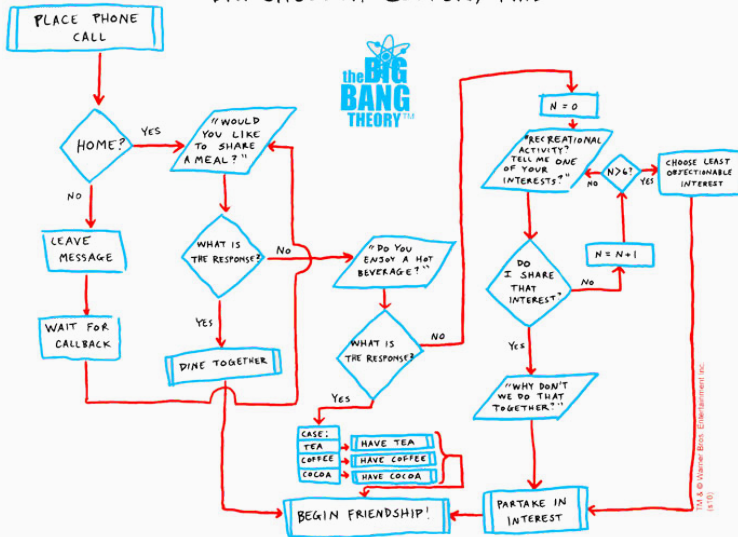
1. Comprensione problema
2. Divide et impera
3. Raffinamenti successivi
4. Scrittura soluzione
5. Test della soluzione

Schemi a blocchi - Costrutti



THE FRIENDSHIP ALGORITHM

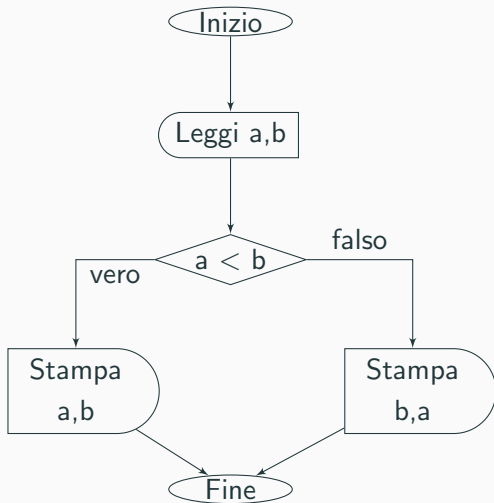
DR. SHELDON COOPER, Ph.D



Esempio 1 - Ordinamento coppie di numeri

Scrivere un programma che, ricevuti in input due numeri a e b , li stampi in ordine dal più piccolo al più grande.

Esempio 1 - Soluzione con schema a blocchi



Esempio 1 - Soluzione in pseudocodice

1. Leggo due numeri a e b
2. Se a è minore di b :
 - 2.1 Stampa su schermo a e b
3. Altrimenti:
 - 3.1 Stampa su schermo b e a
4. Fine

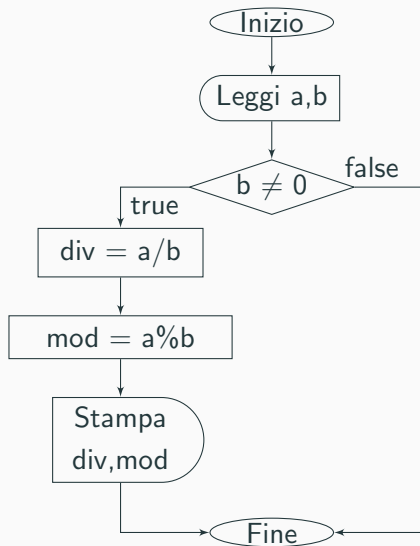
Esercizio 1 - Calcolo della divisione e del resto

Scrivere un programma che, ricevuti in ingresso due numeri a e b , stampi il risultato della divisione a/b ed il relativo resto.

Se l'operazione non è possibile, il programma termina senza stampare alcun messaggio.

Utilizzare l'operatore modulo $\%$ per calcolare il resto della divisione.

Esercizio 1 - Soluzione con schema a blocchi



Esercizio 1 - Soluzione in pseudocodice

1. Leggo due numeri a e b
2. Se b è uguale a zero termino
3. Assegno a div il risultato della divisione intera tra a e b
4. Assegno a mod il resto della divisione intera tra a e b
5. Stampo su schermo div e mod
6. Fine

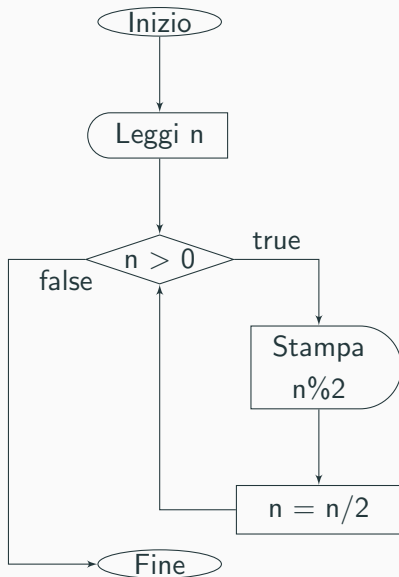
Esercizio 2 - Conversione in binario

Da fare dopo aver visto il binario.

Scrivere un programma che, ricevuto in ingresso un numero positivo n , lo stampi in binario. Si utilizzi il metodo delle divisioni ripetute.

Per semplificare, si stampi separatamente ogni cifra del resto. Si stampi inoltre il risultato in ordine inverso.

Esercizio 2 - soluzione con schema a blocchi



Esercizio 2 - soluzione in pseudocodice

1. Leggi n
2. Ripeti finché n è maggiore di 0:
 - 2.1 Stampa il resto della divisione $n/2$
 - 2.2 Assegna ad n la parte intera della divisione $n/2$
3. Fine

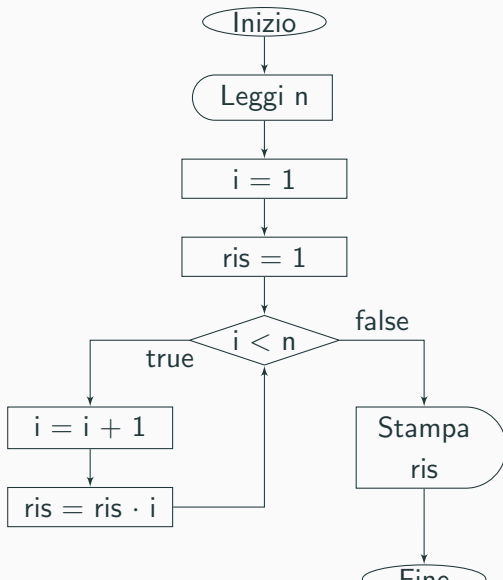
Esercizio 3 - calcolo del fattoriale

Scrivere un programma che, ricevuto in ingresso un numero n , ne calcoli e visualizzi il fattoriale.

Si assuma che l'utente non inserisca mai numeri minori di 1.

Si ricorda che il fattoriale di n è definito come il prodotto di tutti i numeri fra n ed 1: $n! = \prod_{i=1}^n i = 1 \cdot 2 \dots (n-1) \cdot n$

Esercizio 3 - soluzione con schema a blocchi



Esercizio 3 - soluzione in pseudocodice

1. Leggi n
2. Assegna 1 alla variabile i
3. Assegna 1 alla variabile ris
4. Ripeti finché $i < n$:
 - 4.1 Incrementa i di 1
 - 4.2 Assegna a ris il risultato di $ris \cdot i$
5. Stampa il valore di ris
6. Fine

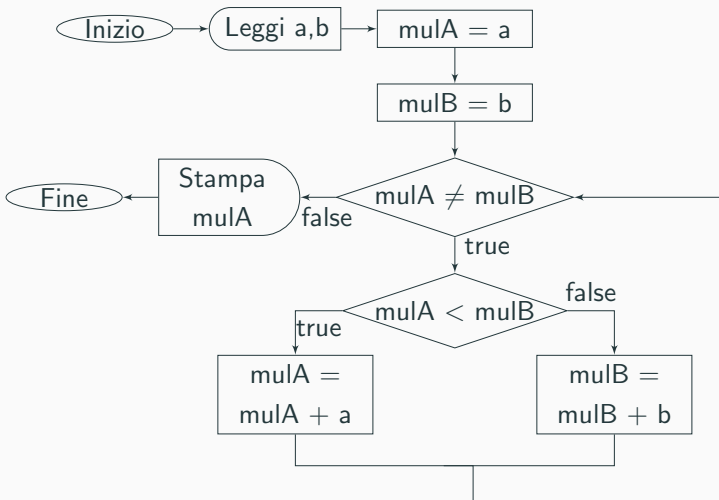
Esercizio 4 - Calcolo del minimo comune multiplo

Scrivere un programma che, ricevuti in ingresso due numeri a e b , ne calcoli e visualizzi il minimo comune multiplo.

Si assuma che a e b siano sempre positivi.

Si ricorda che il minimo comune multiplo è definito come il più piccolo intero positivo multiplo sia di a che di b .

Esercizio 4 - Soluzione con schema a blocchi



Esercizio 4 - Soluzione in pseudocodice

1. Leggi a e b
2. Inizializza $mulA$ e $mulB$ rispettivamente con a e b
3. Ripeti finché $mulA \neq mulB$:
 - 3.1 Incrementa il minore fra $mulA$ e $mulB$ del rispettivo valore (a o b)
4. Stampa $mulA, mulB$
5. Fine

Esercizio 5 - Verifica numero primo

Scrivere un programma che, ricevuto in ingresso un numero n , dica se questo è un numero primo oppure no.

Si assuma che n sia sempre un numero positivo maggiore di 1.

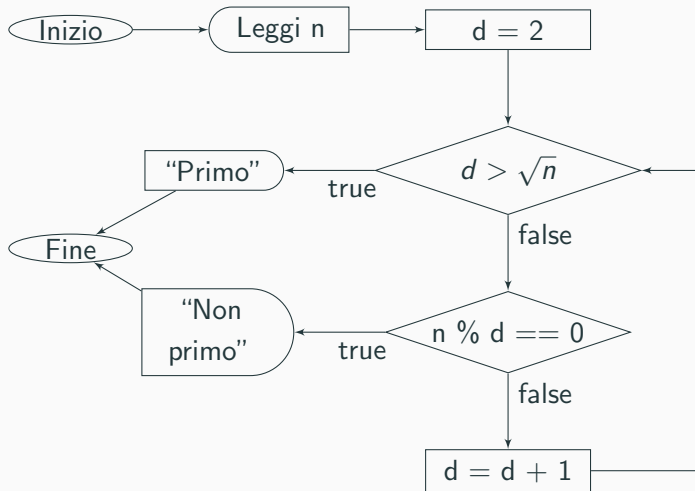
Esercizio 5 - Soluzione in pseudocodice

Cerchiamo un divisore di n :

1. Leggi n
2. Supponi che n sia primo
3. Inizializza d a 2
4. Ripeti finché $d < n$:
 - 4.1 Se d è divisore di n il numero non è primo
 - 4.2 Incrementa d di 1
5. Fine

Tip: non è necessario controllare fino ad n

Esercizio 5 - Soluzione con schema a blocchi



Esercizio 6 - Verifica anno bisestile

Scrivere un programma che, ricevuto in ingresso un anno n , dica se è un anno bisestile.

Un anno è bisestile quando è multiplo di 4. Se è multiplo di 100 non è bisestile, a meno che sia multiplo di 400.

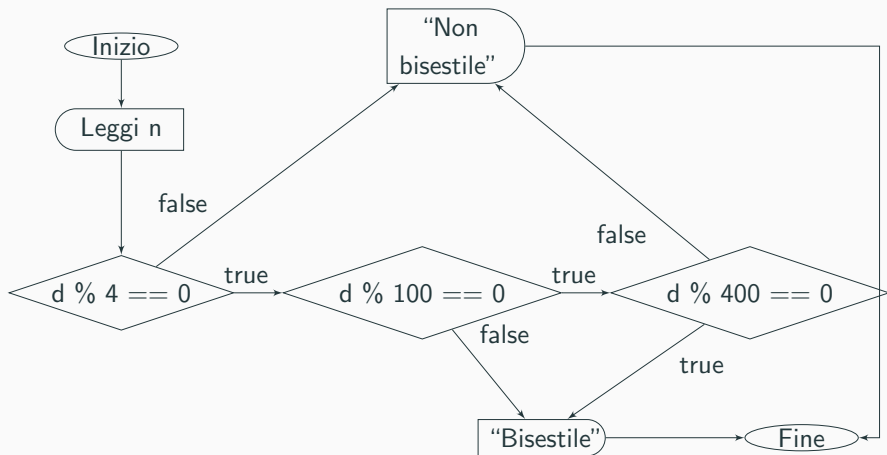
Esercizio 6 - Soluzione in pseudocodice

1. Leggi n
2. Se n non è divisibile per 4 l'anno non è bisestile ed esci
3. Se n non è divisibile per 100 l'anno è bisestile ed esci
4. Se n è divisibile per 400 l'anno è bisestile ed esci
5. L'anno non è bisestile ed esci

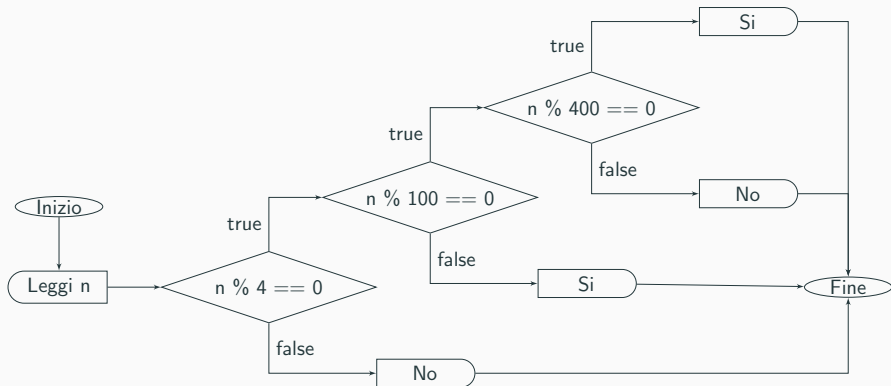
Esercizio 6 - Soluzione in pseudocodice migliore

1. Leggi n
2. Se n è divisibile per 4
 - 2.1 Se n è divisibile per 100
 - 2.1.1 Se n è divisibile per 400 allora è bisestile (4 si, 100 si, 400 si), altrimenti non è bisestile (4 si, 100 si, 400 no)
 - 2.2 Altrimenti è bisestile (4 si, 100 no)
3. Altrimenti non è bisestile (4 no)
4. Fine

Esercizio 6 - Soluzione con schema a blocchi



Esercizio 6 - Soluzione con schema a blocchi migliore



Esercizio 7 - Ricerca per bisezione

Risolvere, solo in pseudocodice, il problema posto a lezione (ricerca efficiente in una sequenza ordinata di elementi).

Idee???

Esercizio 7 - Soluzione in pseudocodice

1. Ricevo una sequenza s di lunghezza l e l'elemento cercato c
2. Se $l == 1$ controllo l'unico elemento della lista e :
 - 2.1 Se $e == c$ allora ho trovato c e termino
 - 2.2 Se $e \neq c$ allora c non esiste in s e termino
3. Altrimenti:
 - 3.1 Considero l'elemento e a metà di s
 - 3.2 Se $e > c$ vado ad 1 utilizzando la prima metà di s come nuova sequenza
 - 3.3 Altrimenti torno ad 1 utilizzando la seconda metà di s (incluso e) come nuova sequenza

Posso aggiungere una condizione 3.4 che controlla direttamente se $e == c$

Se la stringa è di lunghezza pari posso prendere l'elemento immediatamente successivo (o precedente) alla metà esatta

Risolvere in C l'esempio 1 e gli esercizi 1 e 6.