Elementi Di Informatica E Programmazione

Prof. Andrea Loreggia

UNIVERSITÀ DEGLI STUDI DI BRESCIA

UNIVERSITÀ DEGLI STUDI DI BRESCIA

Esercitazione

Esempio: ordinamento del mazzo di carte

UNIVERSITÀ DEGLI STUDI DI BRESCIA

- Problema: Sia dato un mazzo da 40 carte da ordinare in modo che i cuori precedano i quadri, che a loro volta precedono fiori e picche; le carte di uno stesso seme sono ordinate dall’asso al re

Esempio: ordinamento del mazzo di carte

- Problema: Sia dato un mazzo da 40 carte da ordinare in modo che i cuori precedano i quadri, che a loro volta precedono fiori e picche; le carte di uno stesso seme sono ordinate dall’asso al re
- Algoritmo:
  - Si suddivida il mazzo in 4 mazzetti, ciascuno costituito da tutte le carte dello stesso seme
  - Si ordinino le carte di ciascun mazzetto dall’asso al re
  - Si prendano nell’ordine i mazzetti dei cuori, quadri, fiori e picche

Esempio: ordinamento del mazzo di carte

- Problema: Si vuole ricercare, all’interno di un mazzo di chiavi, quella che apre un dato lucchetto

Esempio: ordinamento del mazzo di carte

- Problema: Si vuole ricercare, all'interno di un mazzo di chiavi, quella che apre un dato lucchetto
- Algoritmo:
    1. Si seleziona una chiave dal mazzo
    2. Si marca con un pennarello la chiave e si tenta di aprire il lucchetto con la chiave appena marcata; se funziona, si va al passo 4)
    3. Altrimenti, si controlla la chiave successiva
        1. Se non è marcata, la si marca e si torna al passo 2)
        2. Viceversa, si prende atto che nel mazzo non è presente la chiave che apre il lucchetto
    4. Fine della ricerca

Esempio: radici delle equazioni di 2° grado

- Problema: Calcolo delle radici reali di ax2+bx+c=0

Esempio: radici delle equazioni di 2° grado

- Problema: Calcolo delle radici reali di $ax^2+bx+c=0$
- Algoritmo:
  1. Acquisire i coefficienti a,b,c
  2. Calcolare $\Delta = b^2-4ac$
  3. Se $\Delta<0$ non esistono radici reali, eseguire l'istruzione 7)
  4. Se $\Delta=0$, $x1=x2=-b/2a$, poi eseguire l'istruzione 6)
  5. $x1=(-b+\sqrt{\Delta})/2a$, $x2=(-b-\sqrt{\Delta})/2a$
  6. Comunicare i valori x1, x2
  7. Fine

Esempio: calcolo del M.C.D.

- **Problema**: Calcolare il M.C.D. di due interi a,b, con a>b
- **Algoritmo**: Formalizzato da Euclide nel 300 a.C., si basa sul fatto che ogni divisore comune ad a e b è anche divisore del resto r della divisione intera di a per b, quando a>b e r≠0; se r=0, b è il M.C.D.
  - $MCD(a,b) = MCD(b,r)$, se $r \neq 0$
  - $MCD(a,b) = b$, se $r=0$
- Nota
- L'algoritmo garantisce la determinazione del M.C.D. senza il calcolo di tutti i divisori di a e b

Esempio: calcolo del M.C.D.

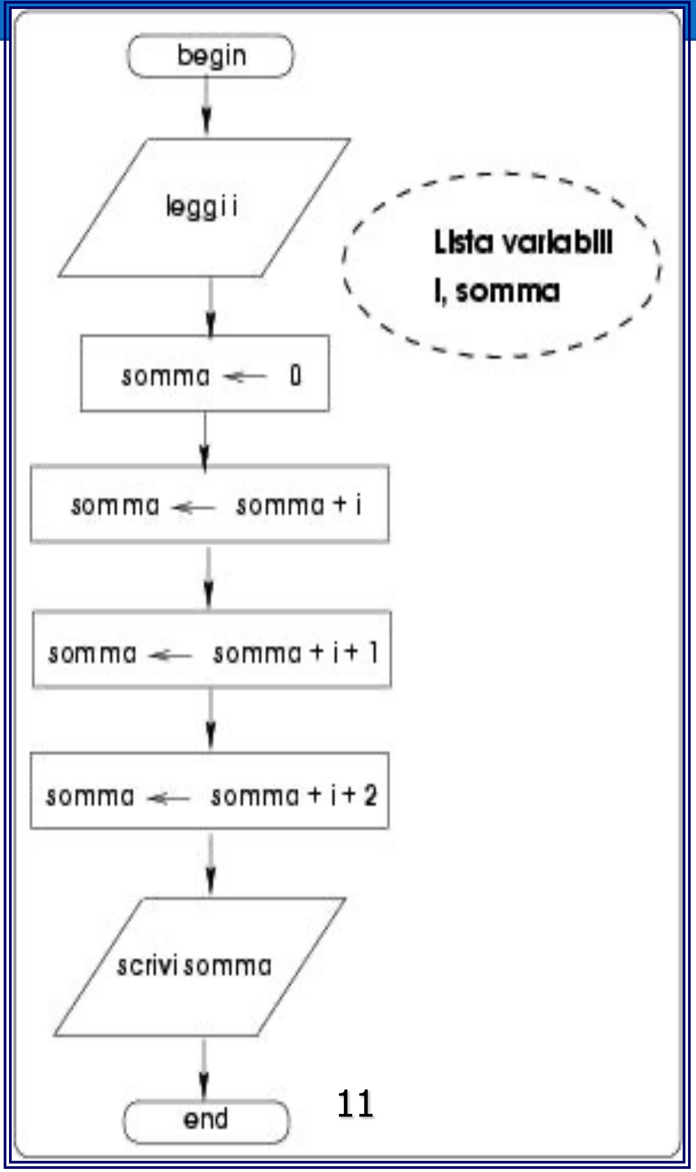
- Acquisire i valori di a e b
- Se $b>a$, scambiare i valori di a e b
- Calcolare il resto r della divisione intera di a per b
- Se $r=0$, $MCD(a,b)=b$; comunicare il risultato all'esterno; eseguire l'istruzione 6)
- Se $r\neq0$, sostituire il valore di a con il valore di b ed il valore di b con il valore di r; tornare al passo 3)
- Fine

Gli algoritmi iterativi

UNIVERSITÀ
DEGLI STUDI
DI BRESCIA

Problema: Calcolare la somma di tre interi consecutivi

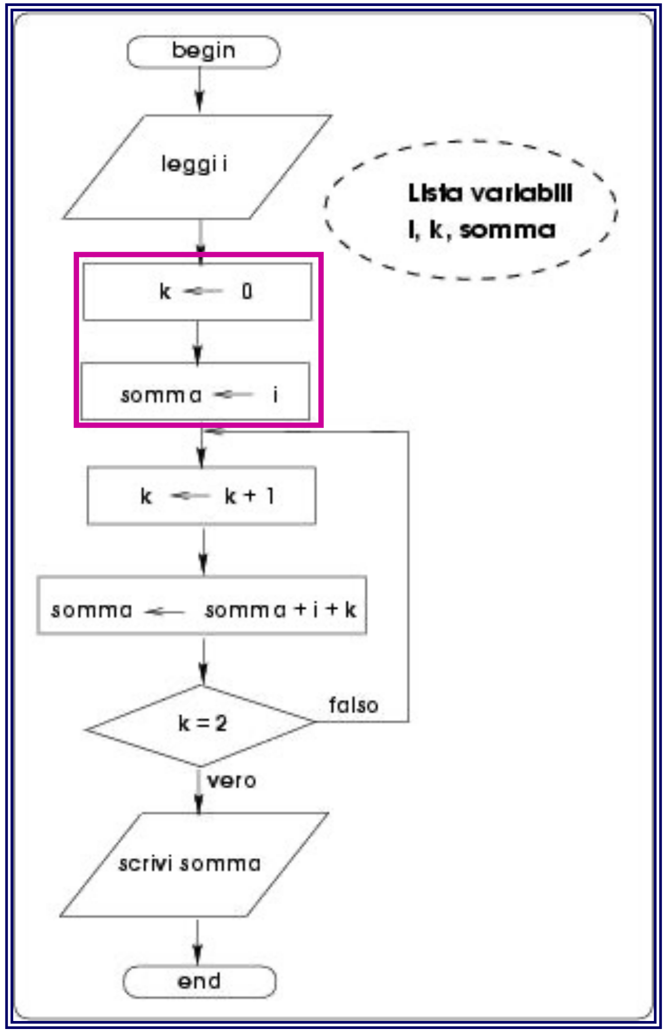
- Note:
  - La variabile somma è un contenitore di somme parziali, finché non si ottiene la somma totale richiesta
  - La soluzione del problema viene raggiunta eseguendo azioni simili per un numero opportuno di volte



Gli algoritmi iterativi

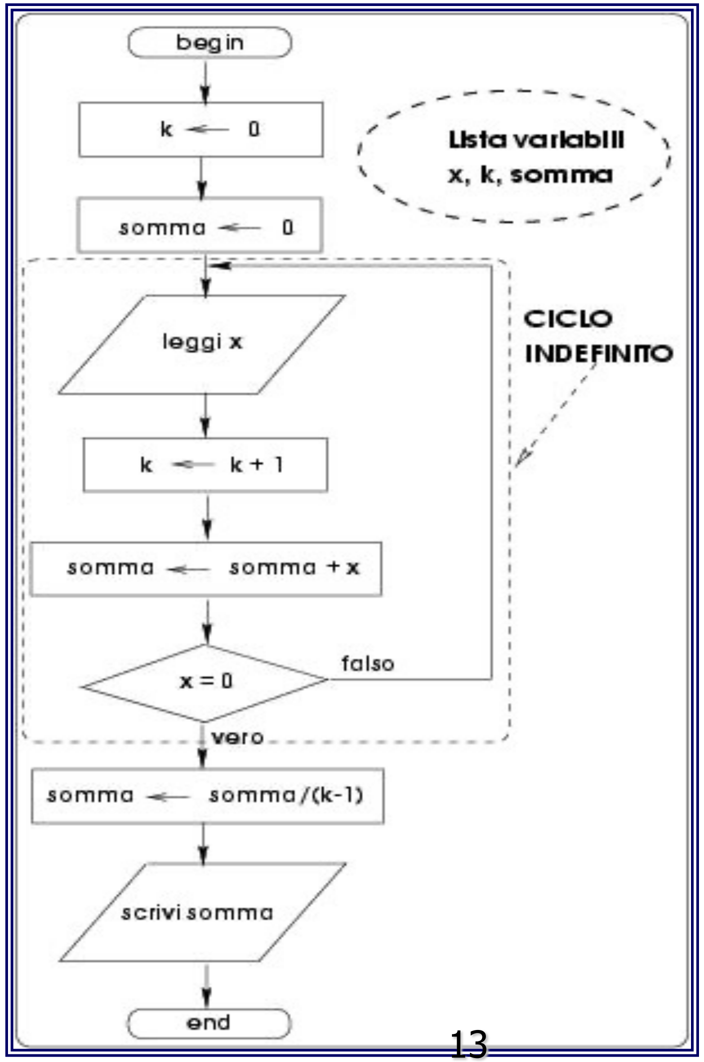
- Problema: Calcolare la somma di tre interi consecutivi

- Note:
  - La fase di inizializzazione riguarda la somma e l'indice del ciclo
  - Il controllo di fine ciclo viene effettuato in coda (iterazione per falso)



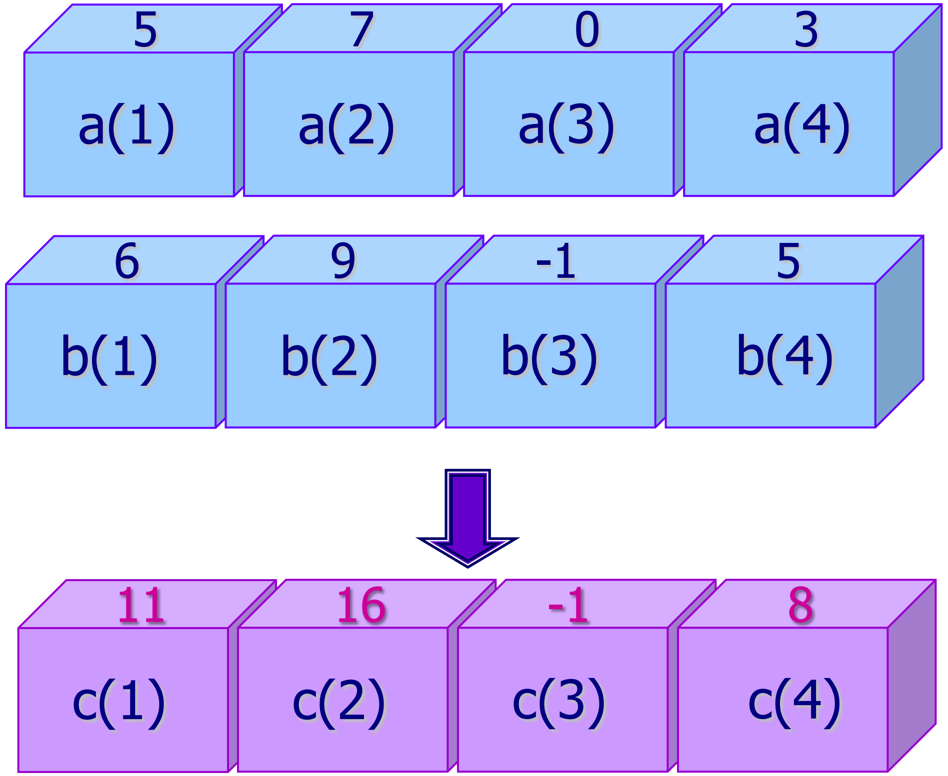
Gli algoritmi iterativi

- Problema: Calcolo della media di un insieme di numeri; non è noto a priori quanti sono i numeri di cui deve essere calcolata la media
  - I numeri vengono letti uno alla volta fino a che non si incontra un $x=0$, che segnala la fine dell'insieme



Gli algoritmi iterativi

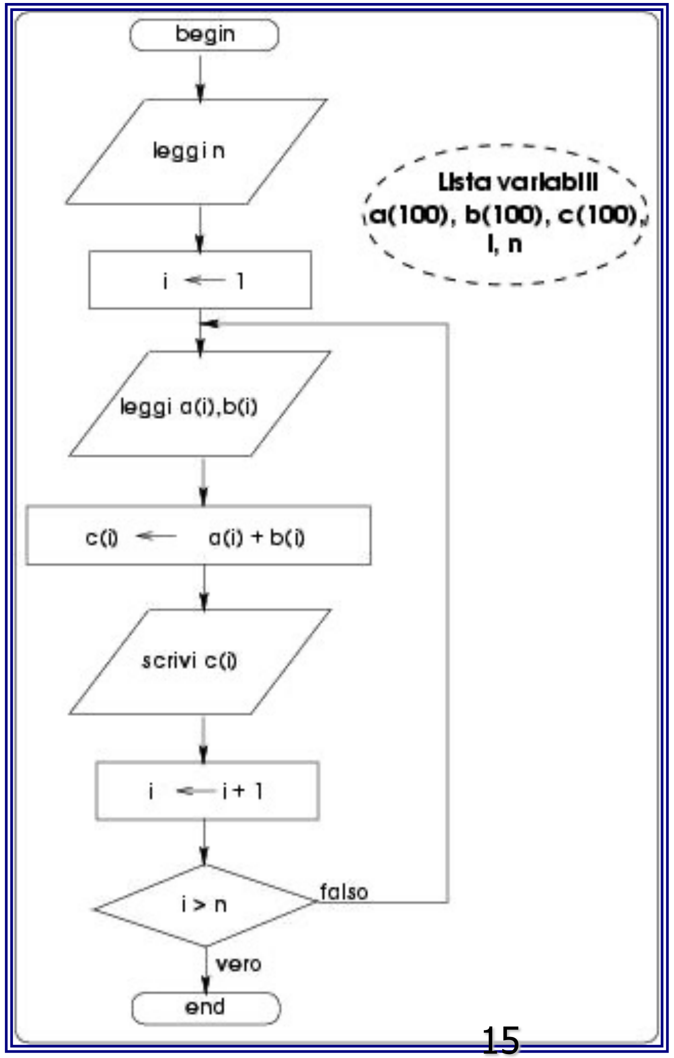
- **Problema:** Calcolare il vettore somma di due vettori di uguale dimensione *n*



Gli algoritmi iterativi

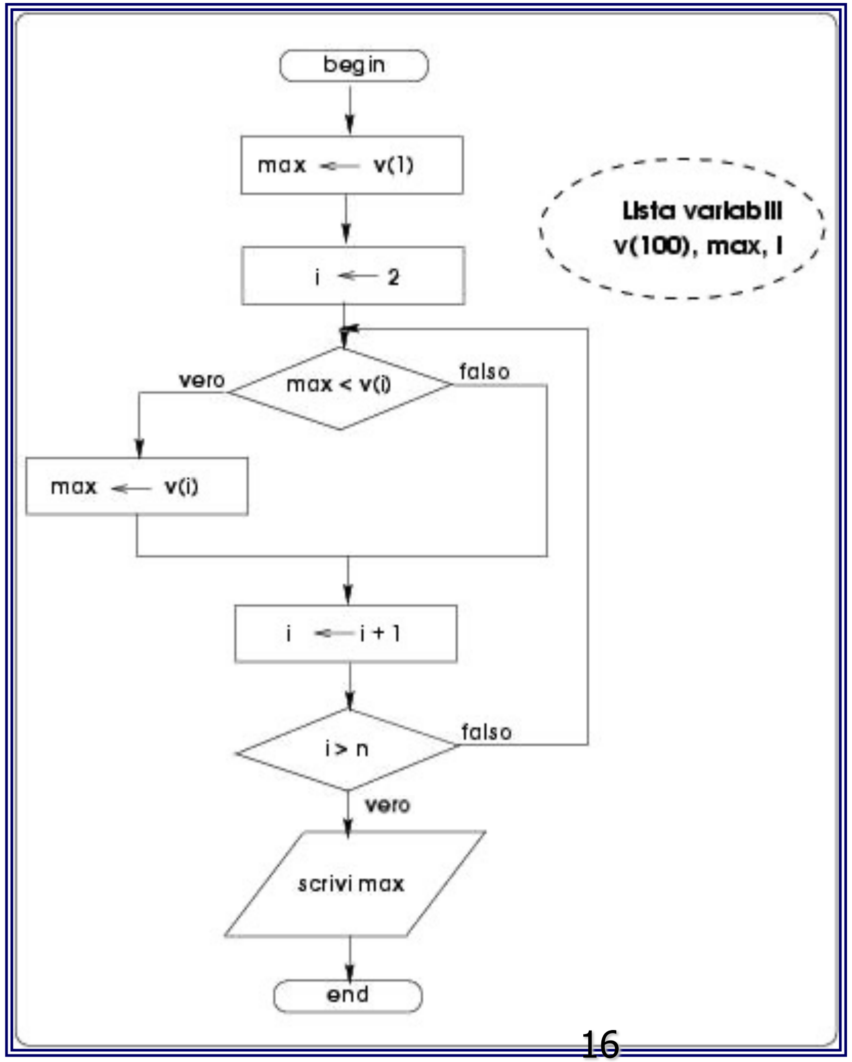
UNIVERSITÀ DEGLI STUDI DI BRESCIA

- L'utilità dei vettori consiste nel poter usare la tecnica iterativa in modo da effettuare la stessa operazione su tutti gli elementi del vettore
- Usando la variabile contatore di un ciclo come indice degli elementi di un vettore è possibile considerarli tutti, uno alla volta, ed eseguire su di essi l'operazione desiderata



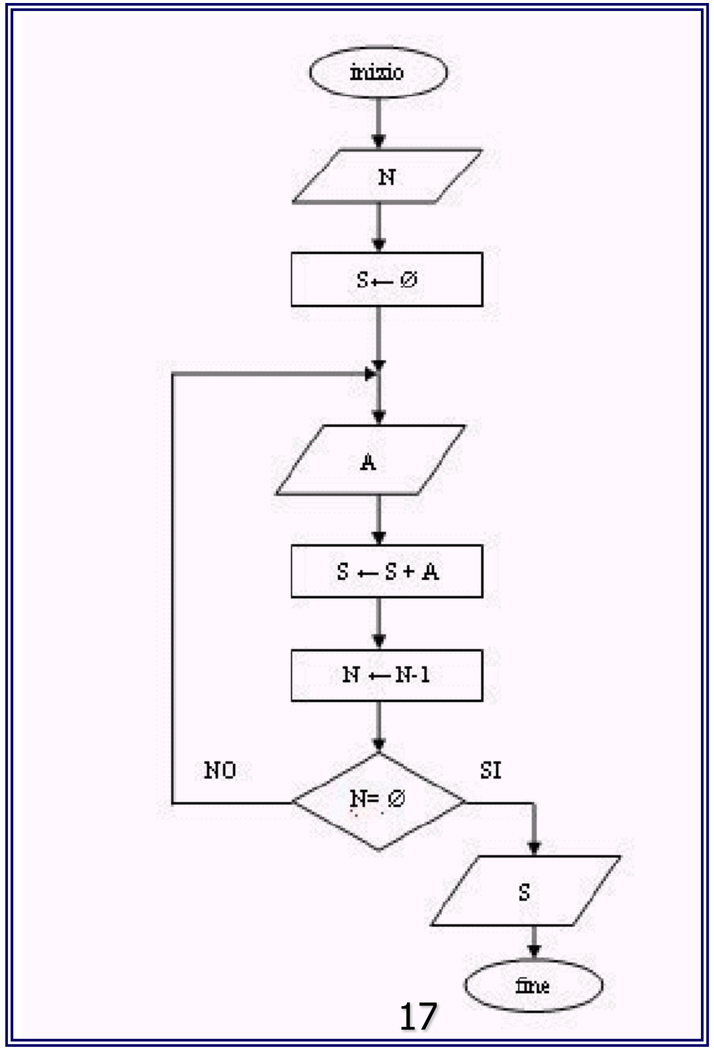
Gli algoritmi iterativi

- Problema: Calcolo del massimo elemento di un vettore



Ancora esempi…

- **Problema:** Somma di una sequenza di numeri
  - Indicando con a_i il generico elemento da sommare, la formula generale è
    - $S = a_1 + a_2 + \ldots + a_n$
  - La variabile *n* conta quante volte si ripete l'iterazione: *n* viene decrementata di 1 ad ogni iterazione ed il ciclo termina quando *n* vale 0
  - La variabile A è usata per l'input degli a_i, S per le somme parziali e totale



Gli algoritmi ricorsivi

- Un algoritmo si dice ricorsivo quando è definito in termini di se stesso, cioè quando una sua istruzione richiede una nuova esecuzione dell'algoritmo stesso
- La definizione ricorsiva di un algoritmo è suddivisa in due parti:

a) la base della ricorsione, che stabilisce le condizioni iniziali, cioè il risultato che si ottiene per i dati iniziali (in generale per 0 e/o 1)

b) la regola di ricorsione, che definisce il risultato per un valore n, diverso dal valore (/i) iniziale per mezzo di un'espressione nella quale si richiede il risultato dell'algoritmo calcolato per $n-1$

Gli algoritmi ricorsivi

UNIVERSITÀ DEGLI STUDI DI BRESCIA

- Gli algoritmi ricorsivi sono particolarmente utili per eseguire compiti ripetitivi su un insieme di input variabili
- L’algoritmo ricorsivo richiama se stesso, generando una sequenza di chiamate che ha termine al verificarsi della condizione di terminazione (che in genere si ha con particolari valori d’ingresso)
- Gli algoritmi ricorsivi sono eleganti e sintetici, ma non sempre efficienti, dato che la ricorsione viene implementata mediante l’uso di chiamate di funzione annidate

Gli algoritmi ricorsivi

- Le chiamate di funzione annidate generano una quantità enorme di overhead, occupando lo stack per un numero di istanze pari alle chiamate della funzione che è necessario effettuare per risolvere un dato problema
  - Possibile lo *stack overflow*
  - Costi computazionali dovuti all'impegno del processore per popolare e distruggere lo stack

Gli algoritmi ricorsivi

- Esempio: Prodotto di numeri interi

$$a \times b = \begin{cases} 0 & \text{se } b=0 \text{ (base della ricorsione)} \\ a \times (b-1)+a & \text{se } b \neq 0 \text{ (regola di ricorsione)} \end{cases}$$

- Secondo la definizione ricorsiva si ha:

$$3 \times 2 = 3 \times 1 + 3 = 3 \times 0 + 3 + 3 = 0 + 3 + 3 = 6$$

- L'esecuzione di un algoritmo ricorsivo termina sempre: la regola di ricorsione prevede nuove esecuzioni su dati decrescenti, fino ad ottenere i dati di inizio ricorsione

Gli algoritmi ricorsivi

- Esempio: Calcolo del fattoriale di un numero intero
  - Il fattoriale di n è il prodotto di tutti gli interi da 1 ad n, cioè

$$n! = n \times (n-1) \times (n-2) \times \ldots \times 2 \times 1$$

  - Per definizione, $0! = 1$

```
begin fattoriale(n)
    if  n = 0
        then r ← 1
        else r ← n × fattoriale(n–1)
    endif
end
```

Esercizio 1

- La successione di Fibonacci
  - Leonardo Pisano, detto Fibonacci, pose il seguente quesito:
    - Una coppia di conigli giovani impiega una unità di tempo a diventare adulta; una coppia adulta impiega una unità di tempo a riprodursi e generare un’altra coppia di conigli (chiaramente giovani); i conigli non muoiono mai
    - Quante coppie di conigli abbiamo al tempo t generico se al tempo $t=0$ non abbiamo conigli e al tempo $t=1$ abbiamo una coppia di giovani conigli?

Esercizio 2

La prima coppia di conigli è data

Esercizio 3

- La successione di Fibonacci
  - Il calcolo di F_n (numero di coppie di conigli), per qualsiasi tempo t, genera la successione dei numeri di Fibonacci
  - La relazione di ricorsione è

$$F_0=0,\ F_1=1,$$
$$F_n = F_{n-1} + F_{n-2}$$

Esercizi

- Scrivere un algoritmo per ogni punto, e rappresentarlo tramite diagramma a blocchi, per la soluzione dei seguenti problemi:
  - calcolare l’area del triangolo
  - trovare il massimo fra due numeri
  - moltiplicare due interi (usando solo l’operazione di somma)
- Formalizzare, tramite diagramma a blocchi, l’algoritmo per…
  - …calcolare le radici reali di equazioni di 2° grado
  - …calcolare il M.C.D. di due numeri con il metodo di Euclide