

Le potenze

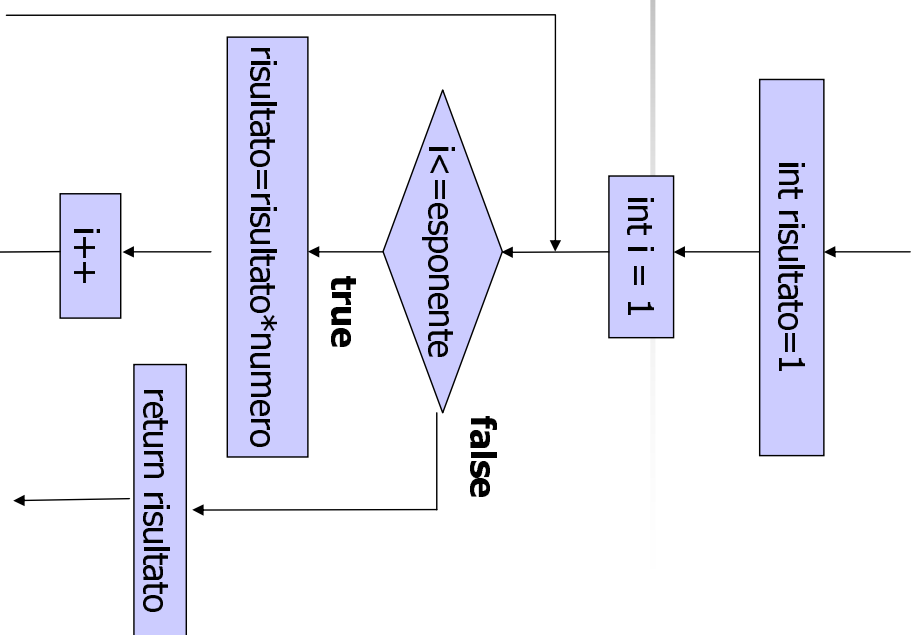
- Implementare un metodo che calcola la potenza di un numero

```
int power(int numero, int esponente)
```

```
int power(int numero, int esponente){  
    int risultato=1;  
    for(int i=1;i<=esponente;i++){  
        risultato=risultato*numero;  
    }  
    return risultato;  
}
```

Le potenze

```
int power(int numero, int esponente){  
    int risultato=1;  
    for(int i=1;i<=esponente;i++){  
        risultato=risultato*numero;  
    }  
    return risultato;  
}
```



Il fattoriale

- Implementare un metodo che calcola il fattoriale

```
int fattoriale(int numero)
```

```
int fattoriale(int numero){  
    int risultato=1;  
    for(int i=1;i<=numero;i++){  
        risultato=risultato*i;  
    }  
    return risultato;  
}
```

Il fattoriale

```
int fattoriale(int numero){
```

```
    int risultato=1;
```

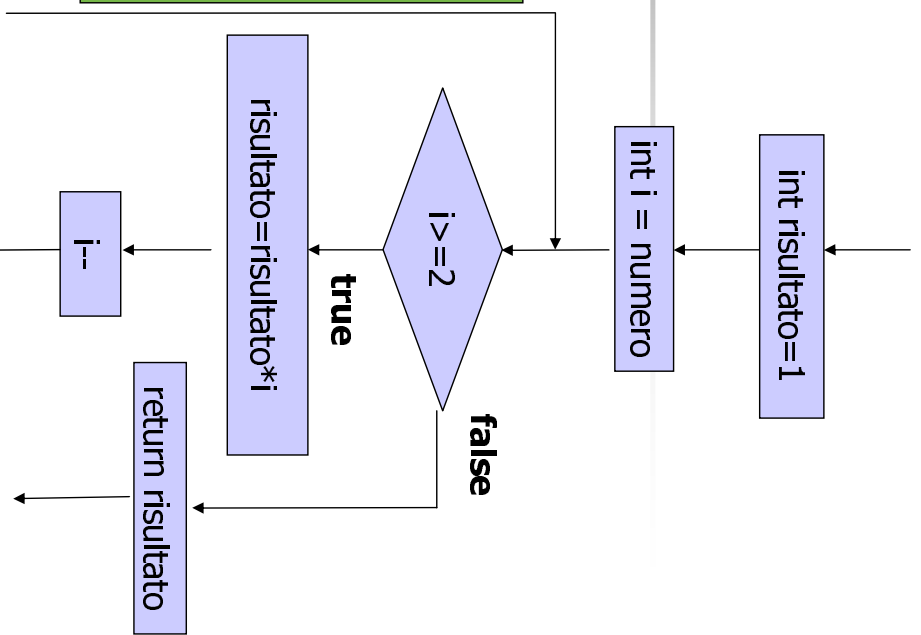
```
    for(int i=numero;i>=2;i--){
```

```
        risultato=risultato*i;
```

```
    }
```

```
    return risultato;
```

```
}
```



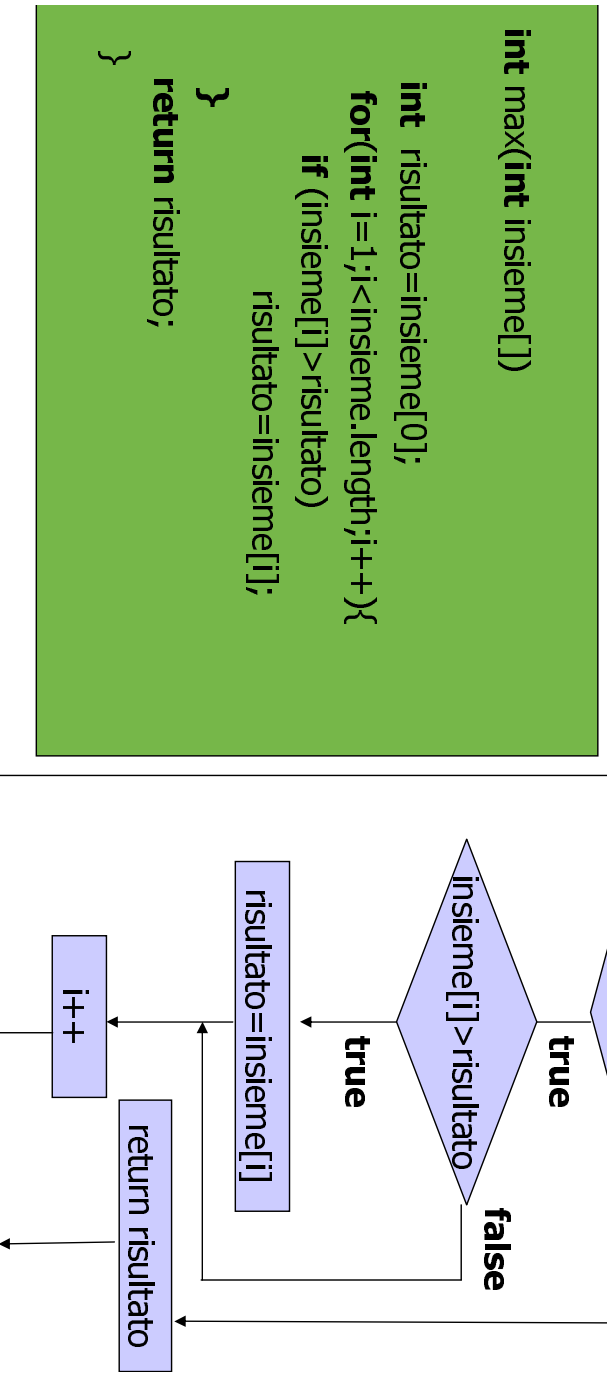
Calcolare il massimo di un insieme

- Implementare un metodo che preso in ingresso un array di interi calcola il massimo
- Si assuma che l'insieme contenga almeno un valore

```
int max(int insieme[])
```

```
int max(int insieme[])  
{  
    int risultato=insieme[0];  
    for(int i=1;i<insieme.length;i++){  
        if (insieme[i]>risultato)  
            risultato=insieme[i];  
    }  
    return risultato;  
}
```

Calcolare il massimo di un insieme



Calcolare il massimo di un insieme II

- Implementare un metodo che preso in ingresso un array di interi calcola il massimo
- Nel caso in cui l'insieme non contenga nessun valore si restituisca Integer.minValue

```
int max(int insieme[])
```

- Integer è una classe predefinita di Java
- max, min, min, minValue variabili che contengono il massimo e il minimo intero rappresentabile

Franco Scarseli

Fondamenti di Informatica 2006-07

7

```
int max(int insieme[])  
int risultato= Integer.minValue;  
for(int i=0;i<insieme.length;i++){  
    if (insieme[i]>risultato)  
        risultato=insieme[i];  
}  
return risultato;  
}
```

Il prodotto dei valori negativi di un insieme

- Implementare un metodo che preso in ingresso un array di interi calcola il prodotto di tutti i valori negativi
- Si assuma che l'insieme contenga almeno un valore

```
int prodottoNegativi(int insieme[])
```

```
int prodottoNegativi(int insieme[])  
int risultato=insieme[0];  
for(int i=1;i<insieme.length;i++){  
    if (insieme[i]<0)  
        risultato=risultato*insieme[i];  
}  
return risultato;  
}
```

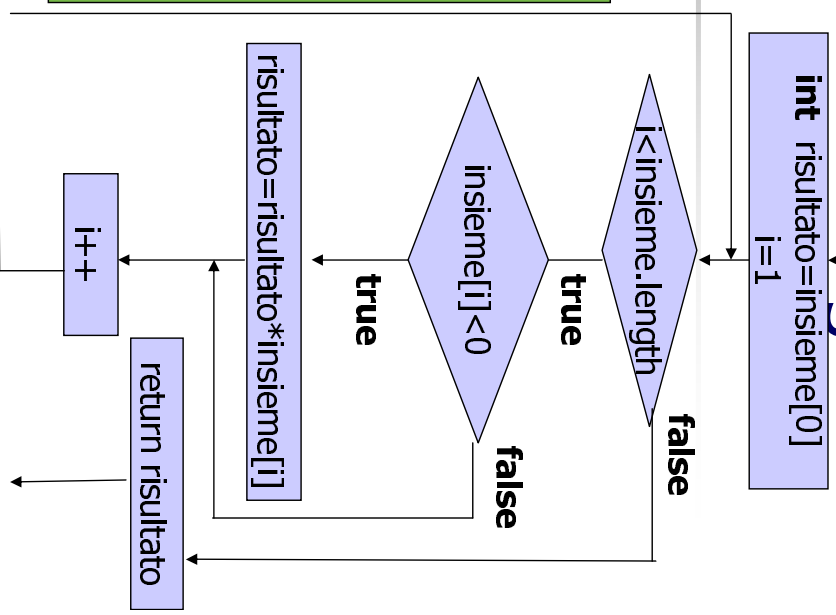
Franco Scarseli

Fondamenti di Informatica 2006-07

8

Il prodotto dei valori negativi di un insieme

```
int prodottoNegativi(int insieme[])  
{  
    int risultato=insieme[0];  
    for(int i=1;i<insieme.length;i++){  
        if (insieme[i]<0)  
            risultato=risultato*insieme[i];  
    }  
    return risultato;  
}
```



Franco Scarseli

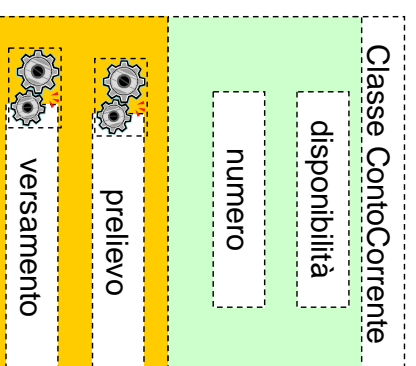
Fondamenti di Informatica 2006-07

9

Implementazione dei metodi: il conto corrente

Il conto corrente

- Si vuole implementare una classe che funzioni come un conto corrente. Il conto corrente ha un numero. Dal conto corrente è possibile versare o prelevare una cifra.
- Per semplicità non importa ricordarsi delle operazioni fatte, ma solo della disponibilità attuale
- Il prelievo deve essere possibile solo se c'è sufficiente disponibilità
- L'oggetto deve rispondere indicando se l'operazione ha avuto successo oppure no



Franco Scarseli

Fondamenti di Informatica 2006-07

10

Implementazione dei metodi: il conto corrente

```
Class ContoCorrente{
String numero;
float disponibilit =0;

ContoCorrente(int n){numero=n;}

String prelievo(float quantita){
if (disponibilit - quantita>=0) {
    disponibilit = disponibilit - quantita;
    return "ok";}
else return "unsuccessful";
}

String deposito(float quantita) {
    disponibilit = disponibilit +quantita;
    return "ok"; }
}
```

Return  
l'istruzione che
definisce il valore
restituito dal
metodo