Titolo nota 22/10/2015

```
int fib ( int n)
{ if (n <= 1) return n;
  else { int i; int succ = 1;
         int prec = 0; int numero;
         for (i = 2; i <= n; i++)
           { numero = prec + succ;
             prec = succ;
             succ = numero;
           }
         return numero;
  }
}
```

$$S_n = S_{n-1} + S_{n-2}$$

3 2 ~~0~~ 1 n ~~1~~ ~~3~~ 5 7

4 ~~3~~ ~~1~~ ~~2~~

n numero intero. Volete controllare se una cifra, cif, compare nelle rappresentazione decimale di n

cif	n	
3	1031 ↑	sì!
3	1001 non compare	NO!

∅ non compare

```
int check ( int n ; int cif )
```

1 cif compare nelle rappr. decimale di n

3234 2

n m interi

n / m quoziente

n % m resto

22 / 3 = 7

22 % 3 = 1

Titolo nota 22/10/2015

```
int dec (int n, int cif)
{
  int trovato = 0;
  while (n > 0 && !trovato)
    if (n % 10 == cif) trovato = 1;
      else n = n / 10;
  return trovato;
}

if (n == 0 && cif == 0) return 1;
else {

}
```

324

324 % 10 = 4

5

32

32 | 0

302 | 0

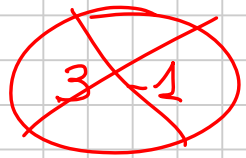
0 | 0

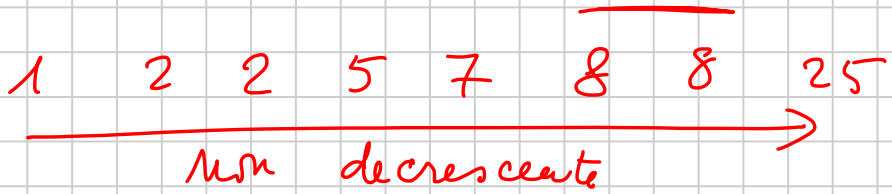
—

—

Titolo nota 22/10/2015

Scrivere una funzione che controlla che un array di interi V sia ordinato in modo non decrescente





0 se l'array è ordinato non

1 se l'array è ordinato

```
int ord (int v[], int dim)
```

Titolo nota 22/10/2015

```
int ord (int v[], int dim)
{
  int i;
  int ordinato = 1;
  for (i = 0; i < dim - 1; i++)
    if (v[i] > v[i+1]) ordinato = 0;

  return ordinato;
}
```

```
for (i = 0; i < dim - 1; i++
  if (v[i] > v[i+1]
    return 0;
```

Titolo nota 22/10/2015

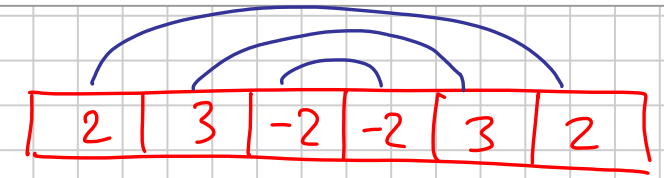
```
int ord (int v[], int dim)
{  int i = 0;
   int ordinato = 1;
-  while (i < dim-1 && ordinato)
      if (v[i] > v[i+1]) ordinato = 0;
         else i = i+1;
   return ordinato;
}
```

(ordinato == 1)

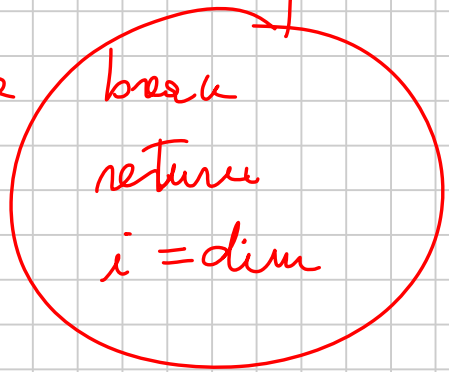
```
while (i < dim)
   if (v[i] <= v[i+1]) { i = dim
      ordinato = 0 }
   else i = i+1;
```

Titolo nota 22/10/2015

scrivere una funzione che controlla che un array sia PALINDROMO



while

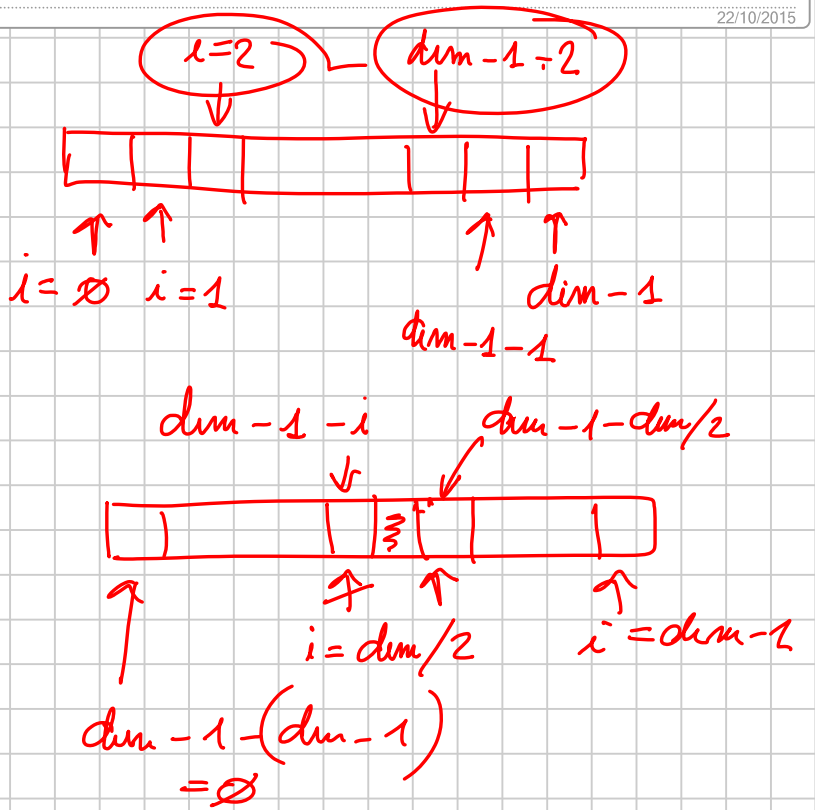



```
int pal (int v[], int dim)
{
```

Titolo nota 22/10/2015

```
int pal (int v[], int dim)
{ int i = 0; int palindromo = 1;
  while (i < dim/2 && palindromo)
    if (v[i] != v[dim-1-i]) palindromo = 0;
       else i = i + 1;

  return palindromo;
}
```



Titolo nota 22/10/2015

```
int pal (int v[], int dim)
{ int j=dim-1;
  int i=0; int palindromo=1;
  while (i<j && palindromo)
    if (v[i] != v[j]) palindromo=0;
      else {i=i+1; j=j-1;}

  return palindromo;
}
```

Successione di Fibonacci

Titolo nota 22/10/2015

$S_0 = 0$ ←

$S_1 = 1$ ←

$S_n = S_{n-1} + S_{n-2}$ per $n > 1$

$S_2 = 1$

$S_3 = 2$

$S_4 = 3$

$S_5 = 5$

$S_6 = 8$

⋮

$(n <= 1)$

if $(n == 0 \;||\; n == 1)$

return n;

```
int fib (int n)                    Sn
{
  if (n == 0) return 0;
  else if (n == 1) return 1;
  else { int i = 2; int prec = 0;
    int corr = 1; int sn;
    while (i <= n)
      {sn = prec + corr;
       prec = corr;
       corr = sn;
      } i = i + 1;
    return sn;
}
```