



Problema: calcolare il massimo tra K numeri

- Scrivere un algoritmo che fornisca in input ad un programma un numero K e K interi positivi. L'algoritmo deve restituire il valore massimo tra quelli introdotti e stampare in ordine inverso i numeri inseriti.
- Sembra un problema già visto:
 - Si supponga di fornire in input ad un programma un numero K e K interi positivi. Il programma deve restituire il valore minimo tra quelli introdotti.
- ma non lo è!! Infatti dobbiamo memorizzare tutti i valori inseriti per stampare il primo inserito per ultimo
- Ma quante variabili usiamo?



Gli array

- Nelle situazioni in cui si devono memorizzare un INSIEME di dati allora si può usare la struttura di dati chiamata **vettore** o **array**
- Un array (vettore) è costituito da una sequenza di elementi consecutivi nella memoria di un calcolatore
- Un array si può vedere come una generalizzazione del concetto di variabile

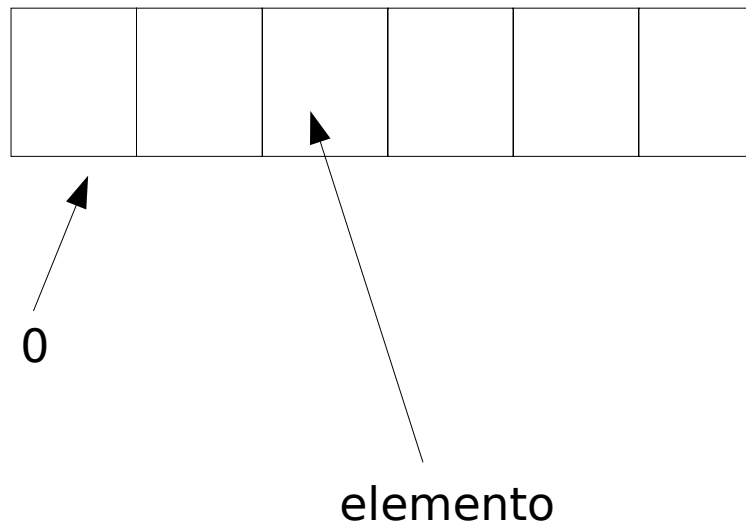


Gli array

- Ad ogni istante un array individua un insieme di valori
- Ogni singolo valore nell'array è individuato dall'identificatore (nome) dell'array e da un indice che ne individua la posizione nella sequenza degli elementi nell'array stesso
- Un array è caratterizzato da una dimensione che esprime il numero di elementi che contiene
- L'operazione di assegnazione permette di modificare il valore di un determinato elemento della sequenza come per una qualsiasi variabile
- In genere gli elementi di un array sono tutti dello stesso tipo (stringhe, interi, caratteri, ecc.) ma in alcuni linguaggi di programmazione sono permessi array di elementi di tipo diverso, es. JavaScript, ma non è una scelta consigliabile

Gli array in Javascript

- Gli array (o vettori) contengono un insieme di dati rappresentati da un singolo nome di variabile
- Un array contiene un insieme di elementi
- Ogni elemento è identificato da un indice (il primo ha indice zero)





Gli array in Javascript

- Un array è creato mediante la seguente dichiarazione:

```
var nome_array = new Array (numero_di_elementi)
```

- Si fa riferimento ad un elemento, ad esempio, nei seguenti modi:

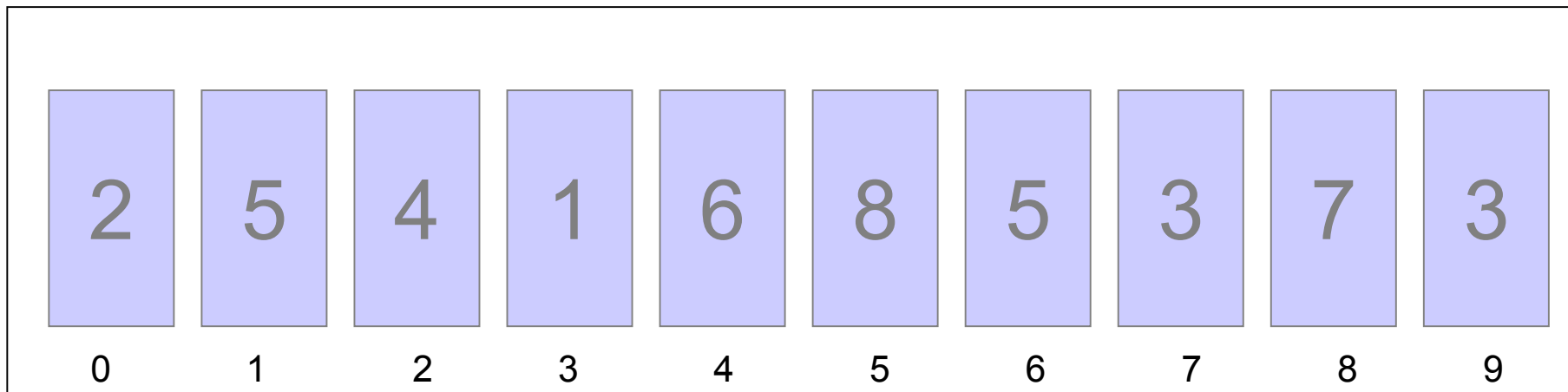
- *nome_array[0] = "Pippo";*
- *nome_array[y] = 27*sqrt(k/2);*
- *window.prompt(nome_array[k*3]);*



Problema: calcolare il massimo tra K numeri

- Ipotizziamo di aver già letto i numeri

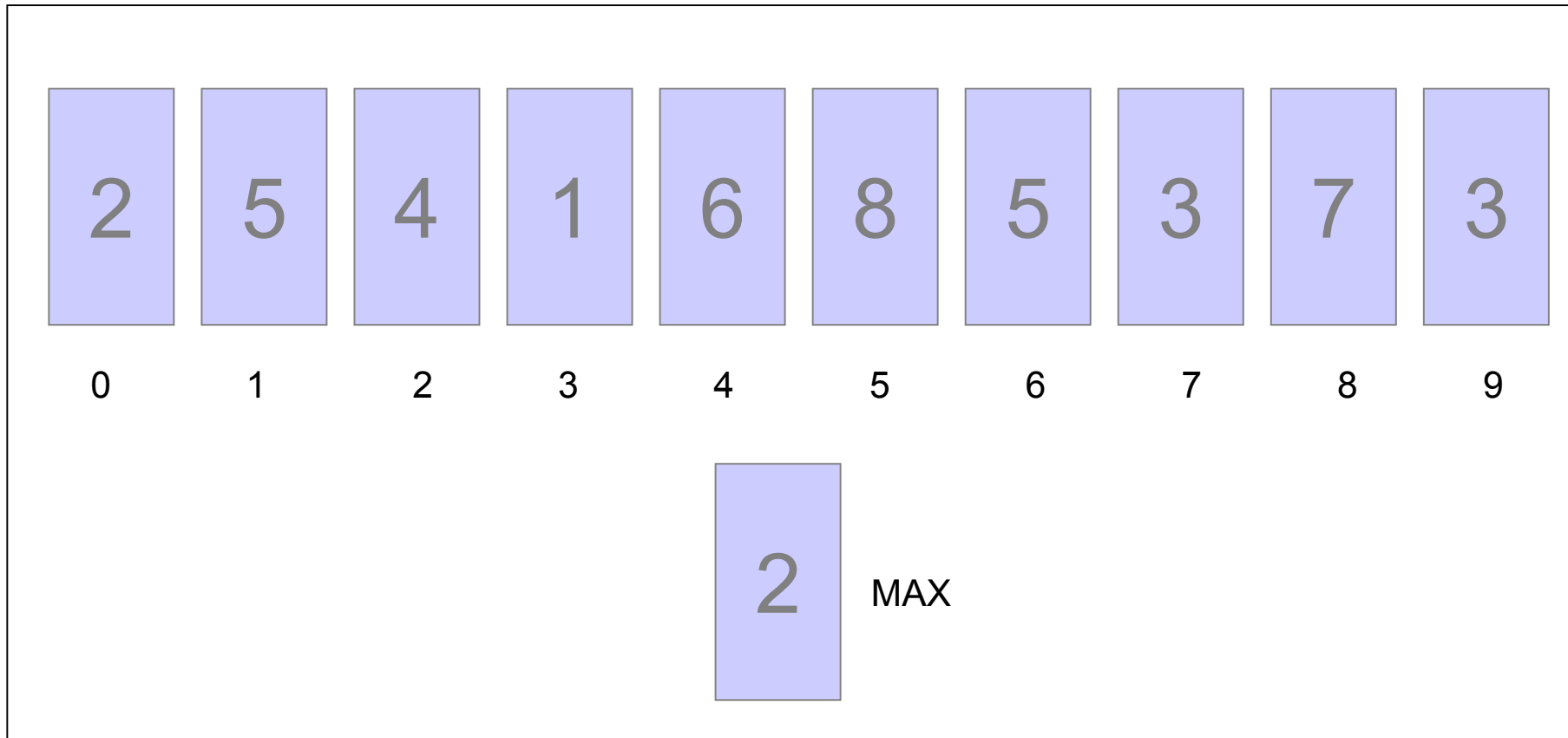
ListaNumeri





Problema: calcolare il massimo tra K numeri

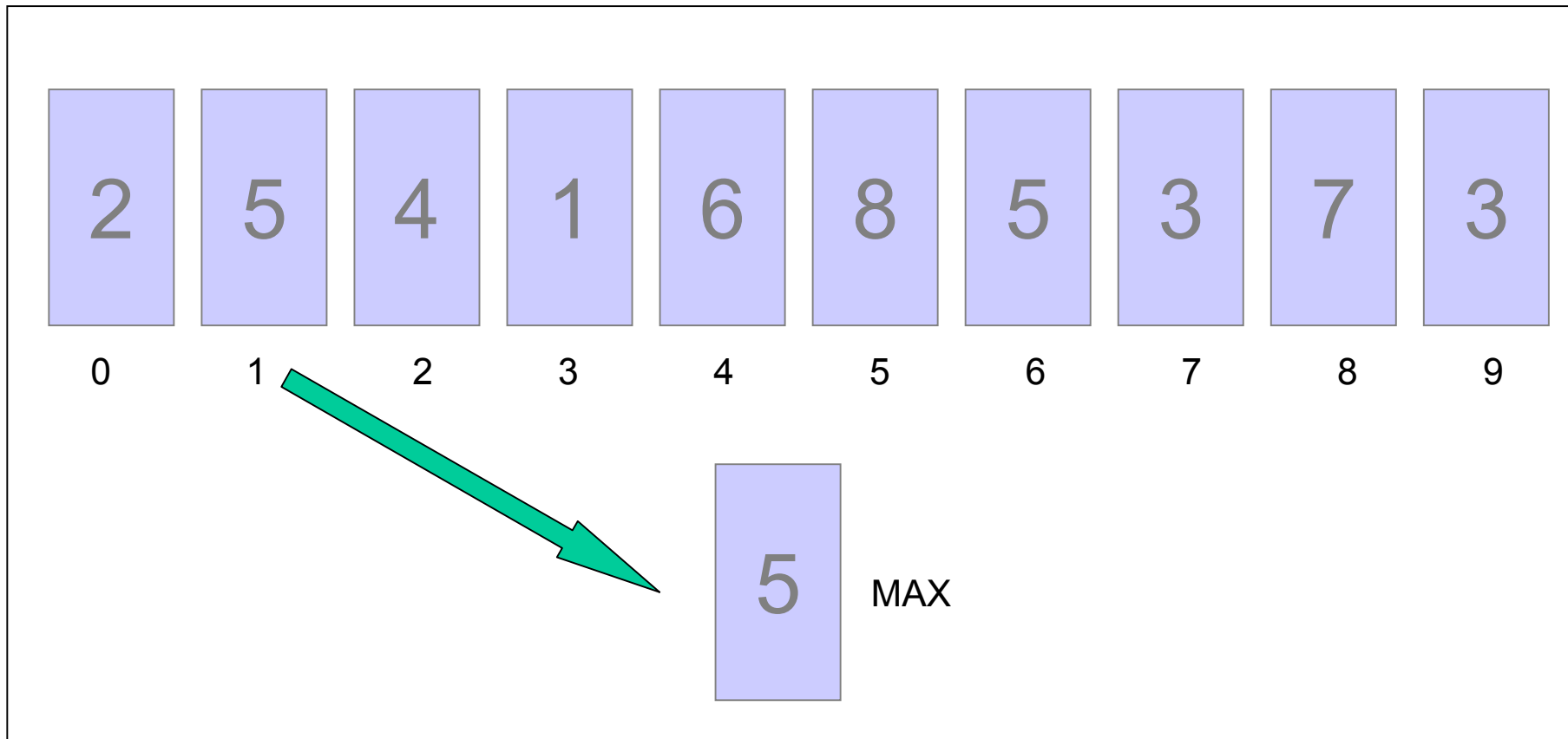
ListaNumeri





Problema: calcolare il massimo tra K numeri

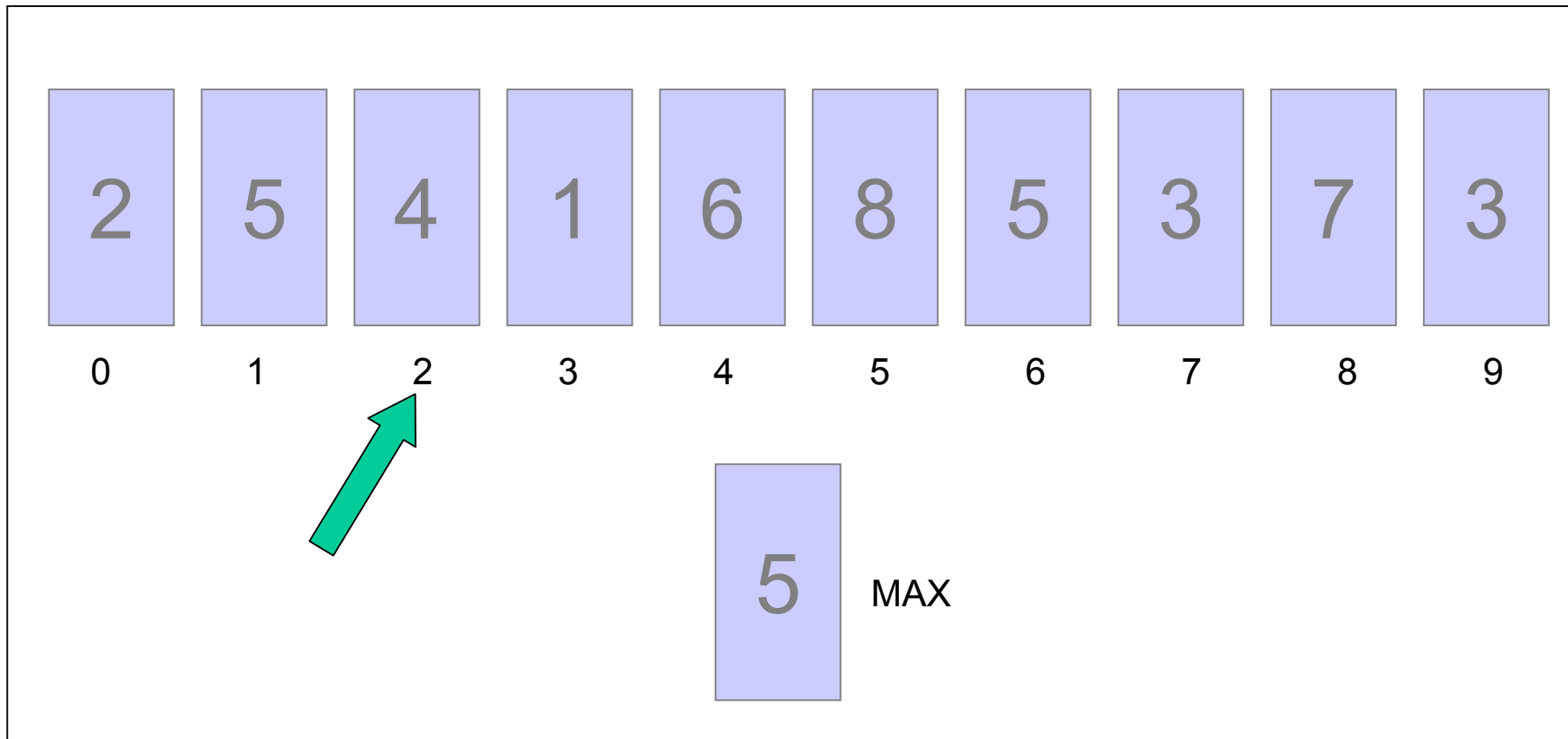
ListaNumeri





Problema: calcolare il massimo tra K numeri

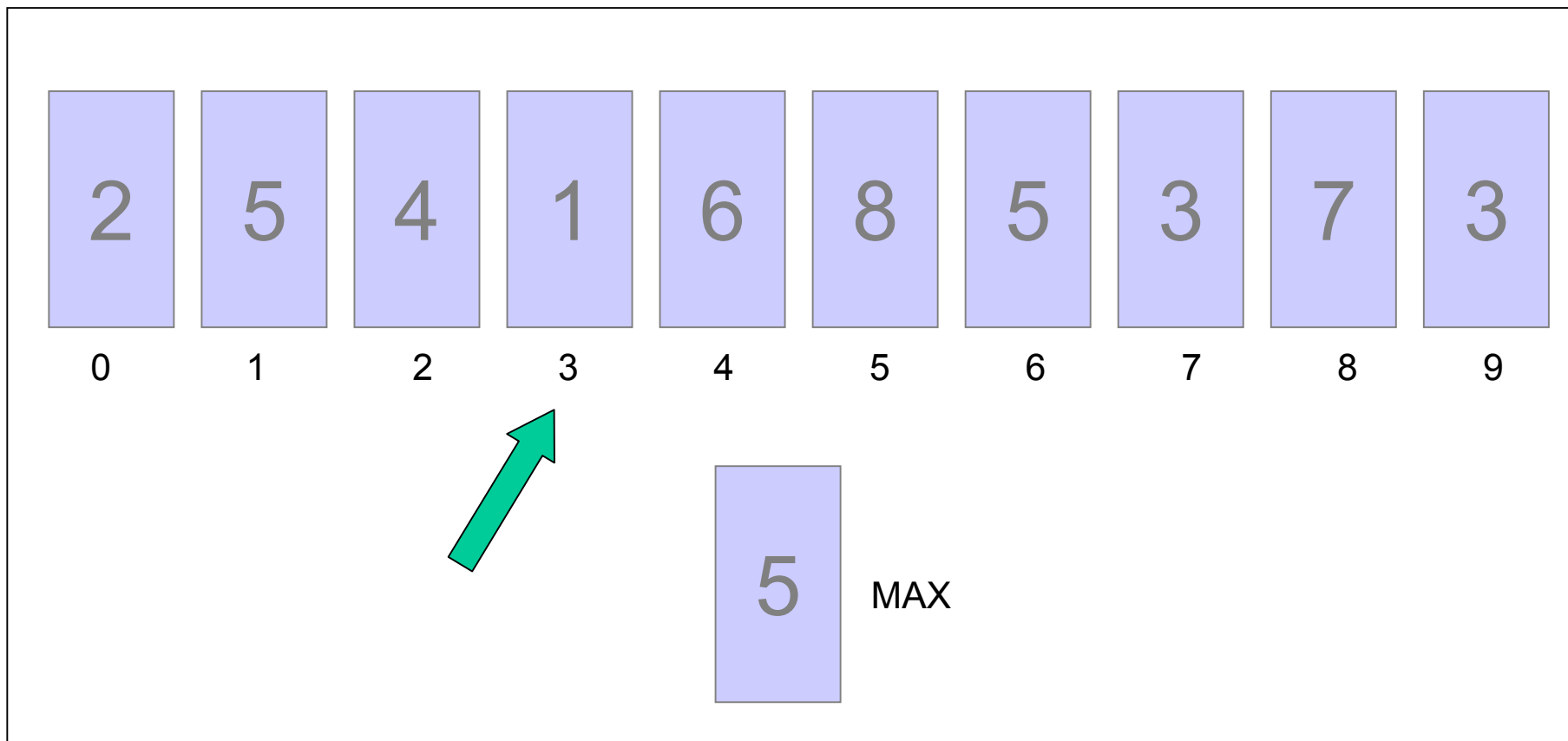
ListaNumeri





Problema: calcolare il massimo tra K numeri

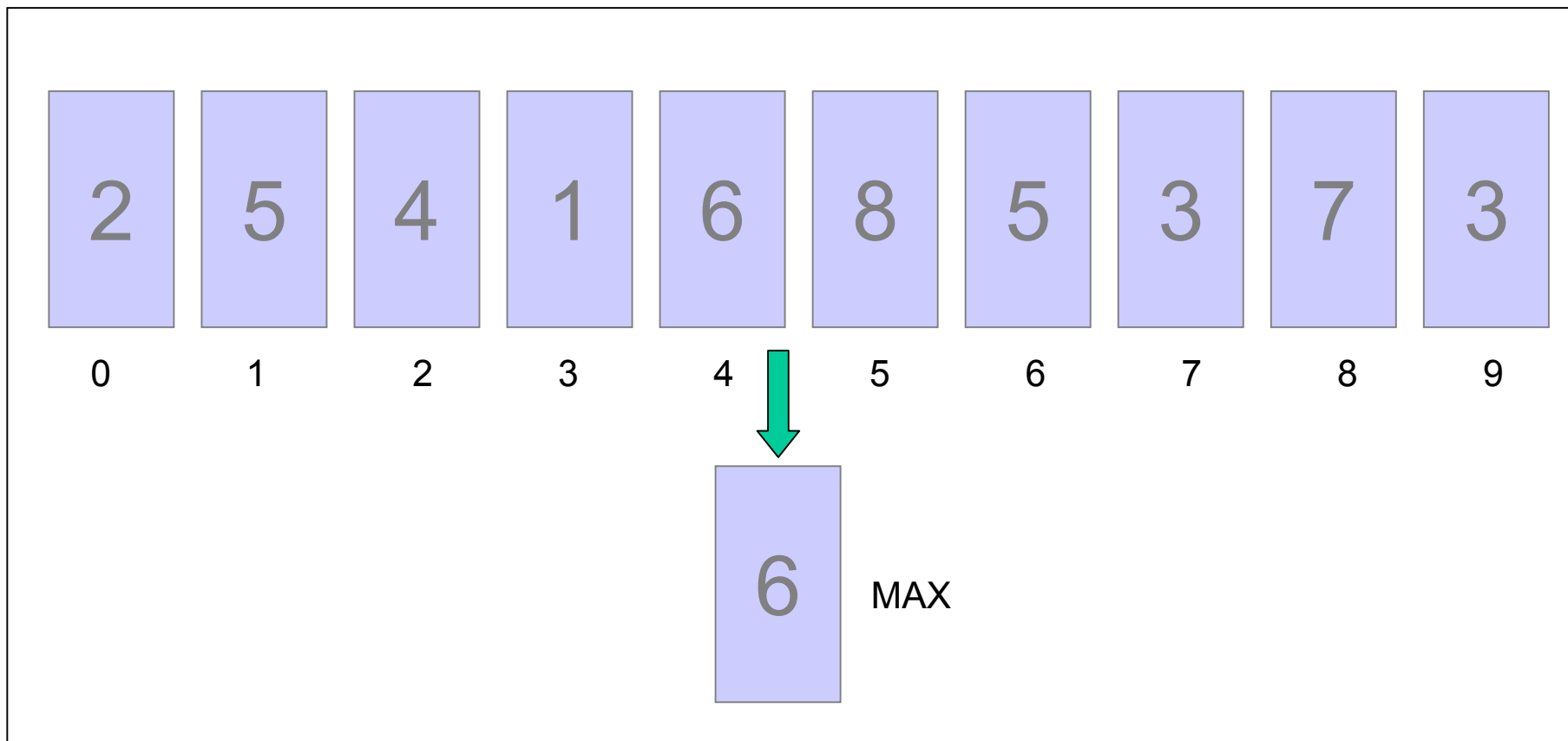
ListaNumeri





Problema: calcolare il massimo tra K numeri

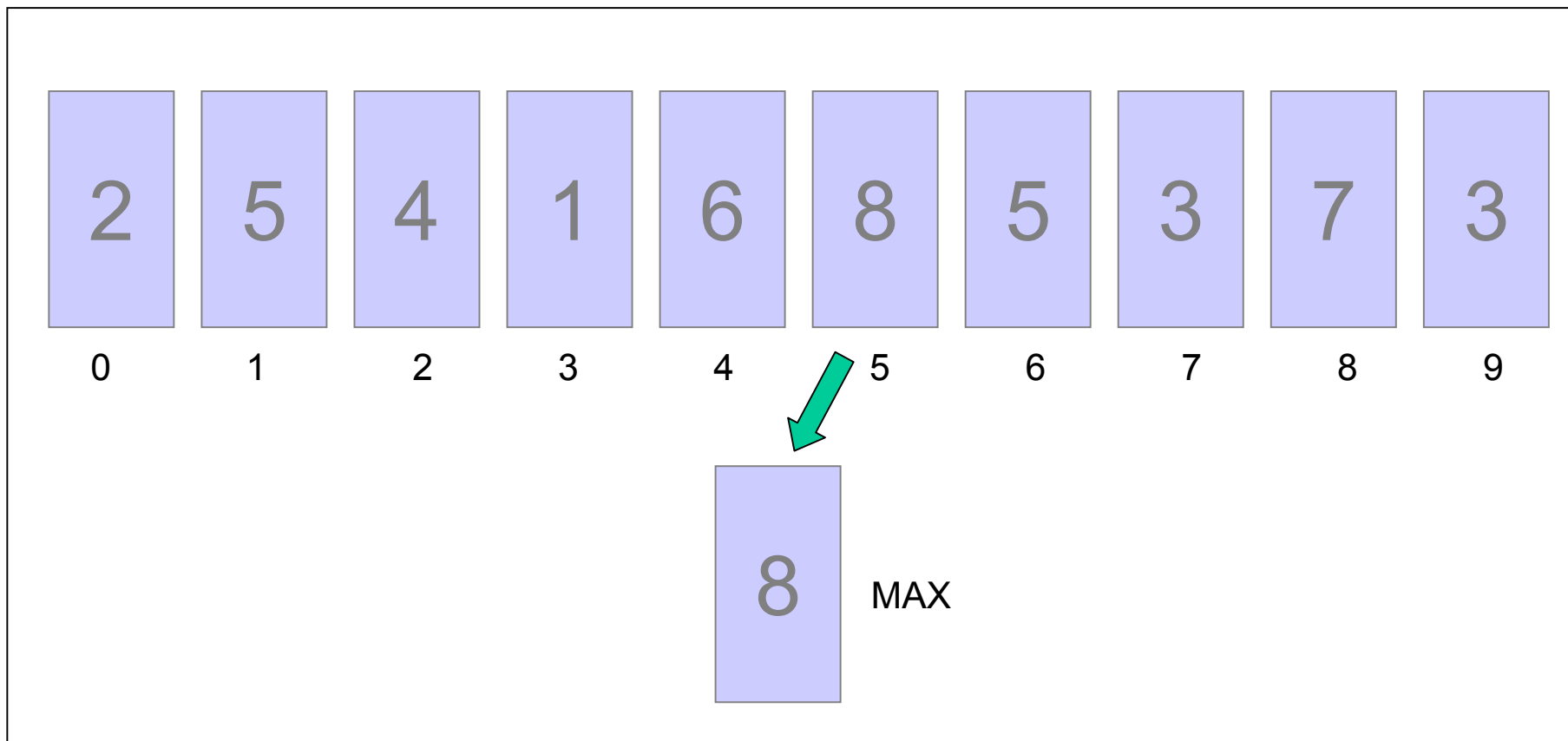
ListaNumeri





Problema: calcolare il massimo tra K numeri

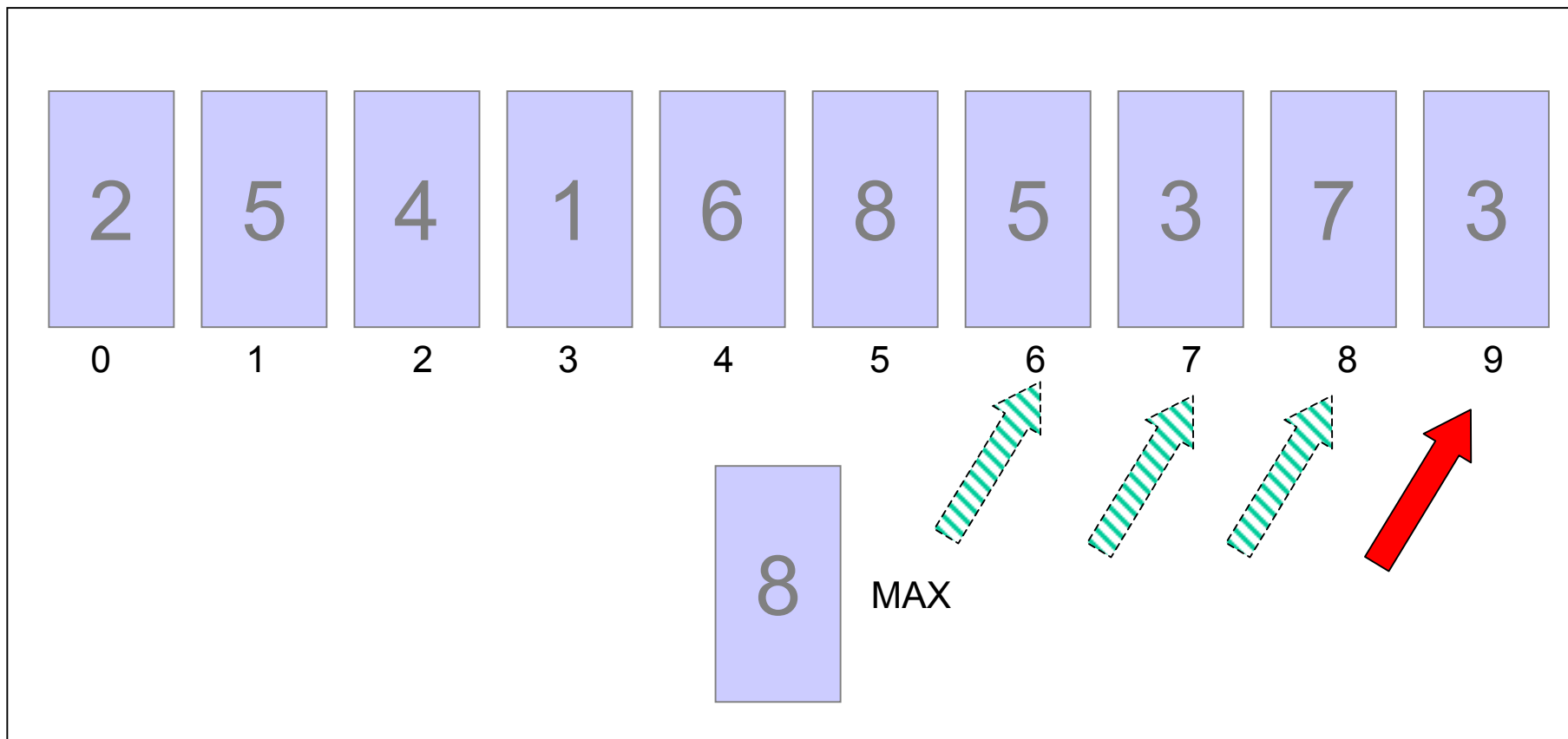
ListaNumeri



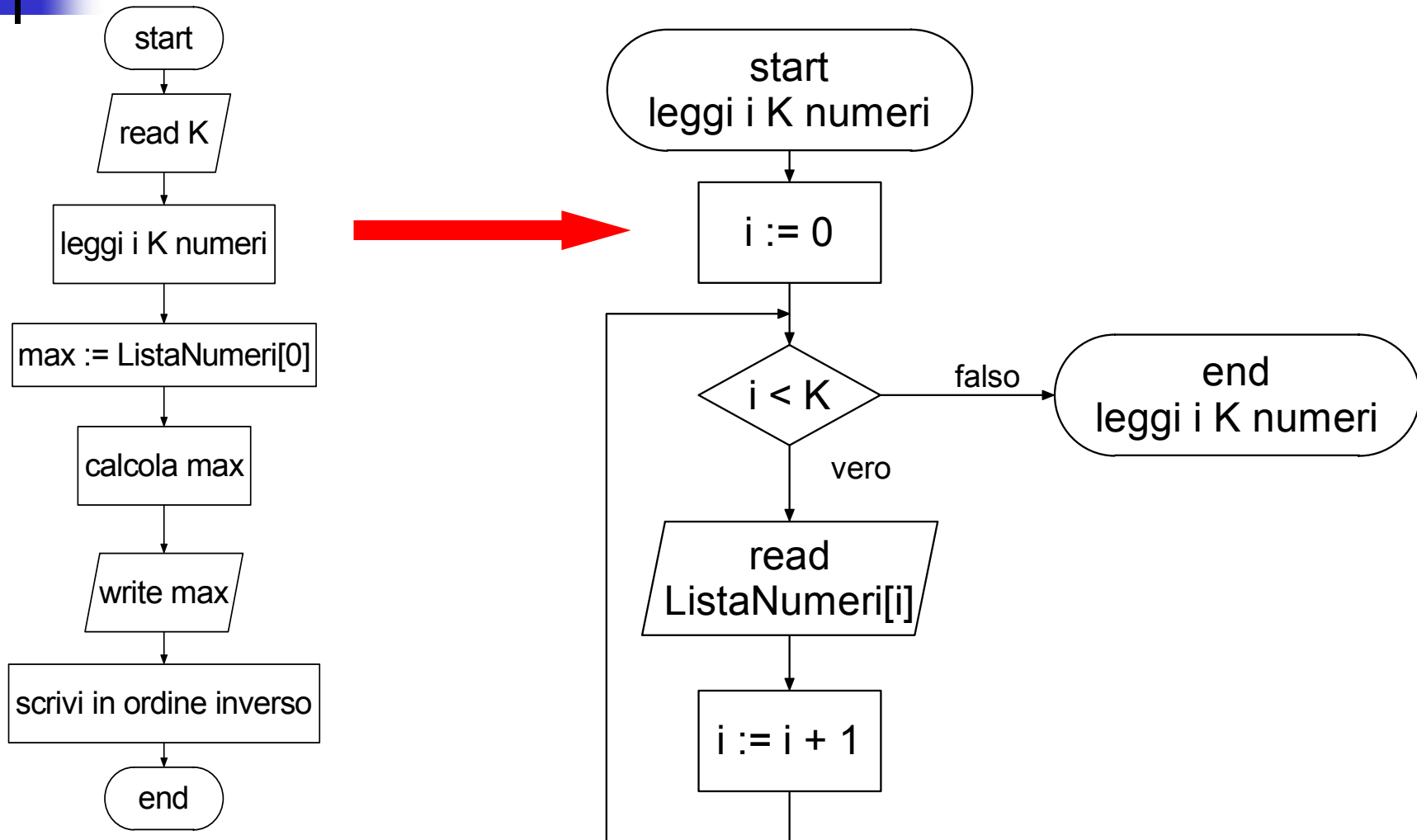


Problema: calcolare il massimo tra K numeri

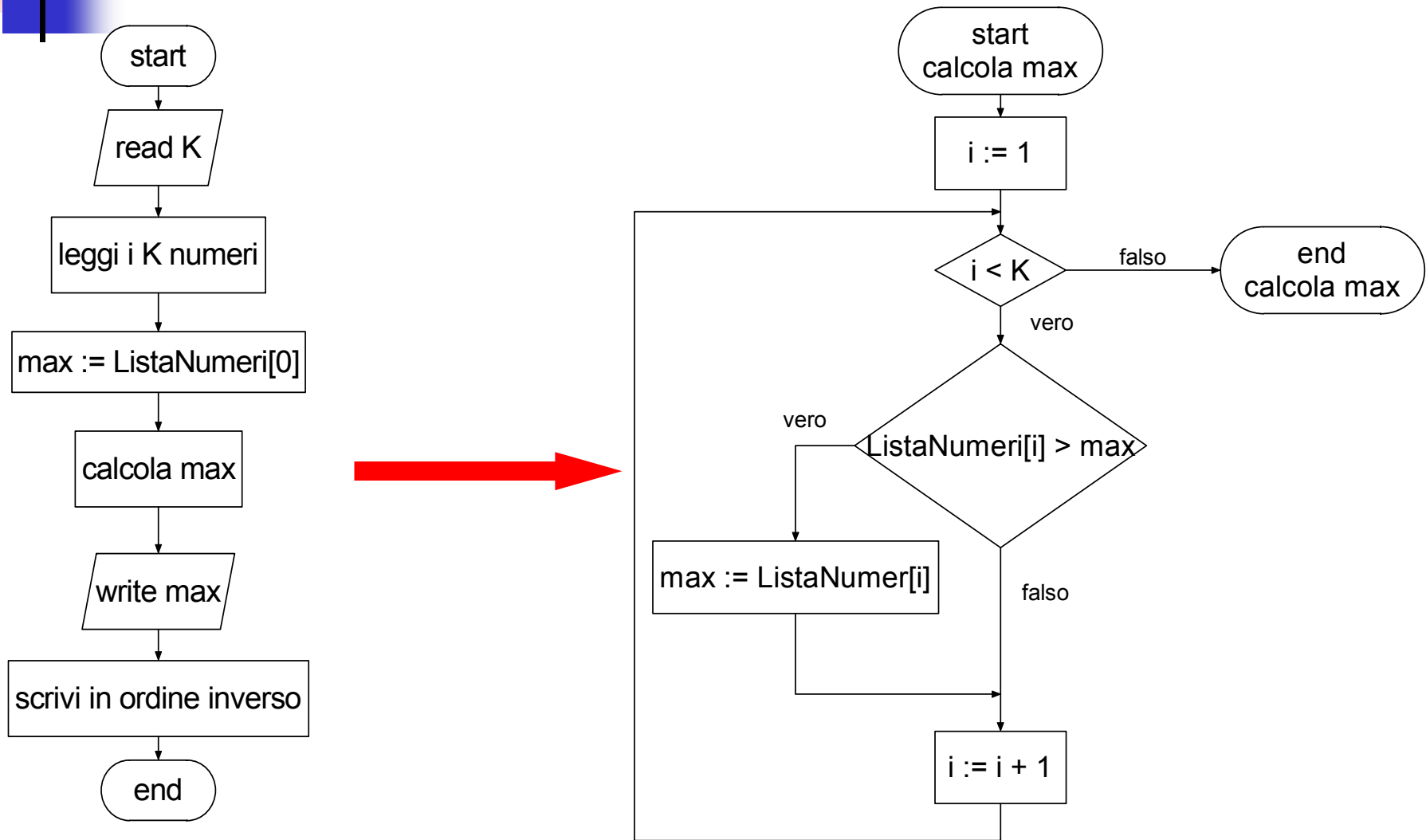
ListaNumeri



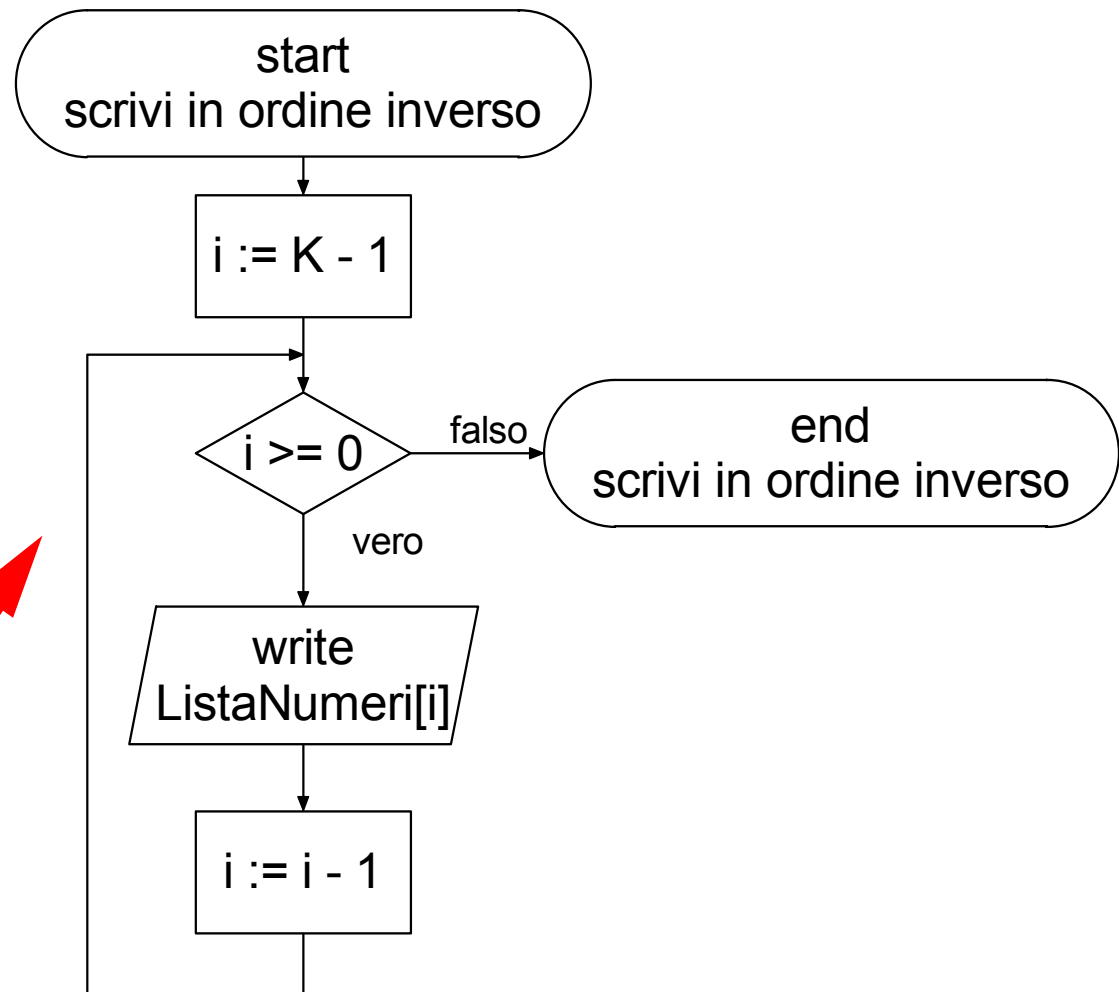
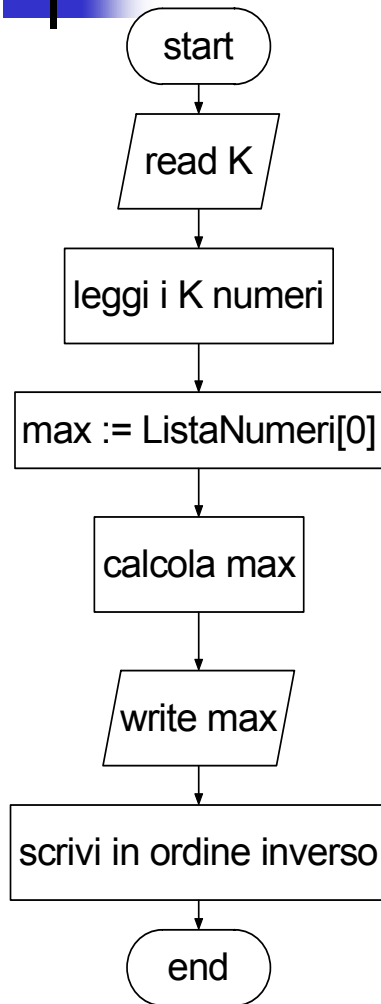
Problema: soluzione preliminare e raffinamenti



Problema: soluzione preliminare e raffinamenti



Problema: soluzione preliminare e raffinamenti





Esercizio

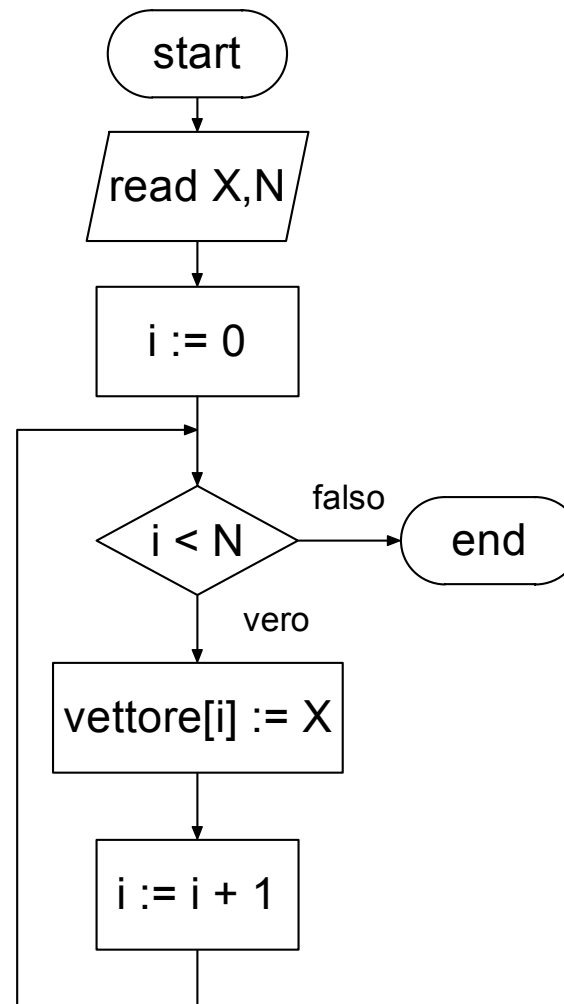
- Tradurre il precedente algoritmo e tutti i successivi in pseudo-codice
- Scrivere il codice Javascript che traduca l'algoritmo



Esempio

- Assegnare a tutte le posizioni di un vettore un numero X . Assumere che la dimensione del vettore sia uguale ad N e che la prima posizione del vettore sia uguale a 0

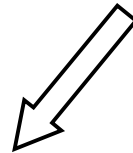
Esempio



Codice Javascript

```
<html>
<head>
  <title>Esercizio sugli array</title>
</head>
<body>
<script>
<!-- Inizio script JavaScript
var N,X,i;
X = window.prompt("Inserisci il numero X");
X = X * 1;
N = window.prompt("Inserisci la lunghezza dell'array");
N = N * 1;
var vettore = new Array(N);
i = 0;
while (i < N)
{
  vettore[i] = X;
  i = i + 1;
}
i = 0;
while (i < N)
{
  window.alert("vettore[" + i + "]=" + vettore[i]);
  i = i + 1;
}
// Fine script -->
</script>
</body>
</html>
```

Inserito per mostrare
il contenuto dell'array

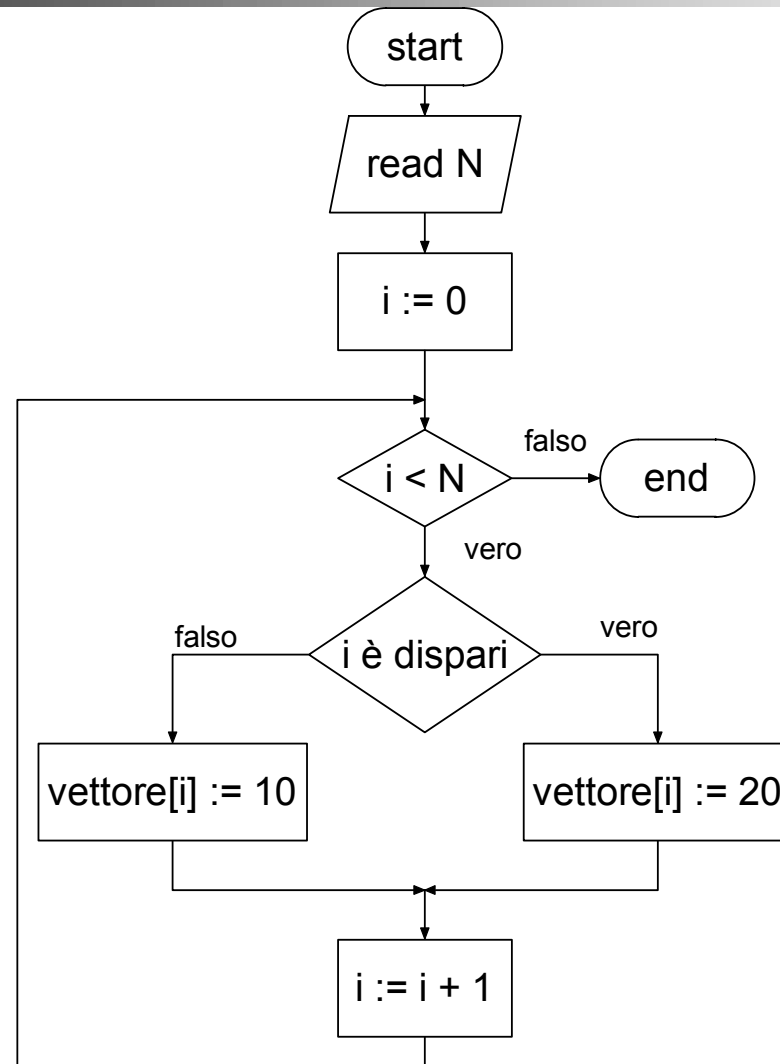




Esempio

- Assegnare alle posizioni di indice pari di un vettore il valore 10 e alle posizioni dispari il valore 20. Assumere che la dimensione del vettore sia uguale ad N e che la prima posizione del vettore sia uguale a 0

Esempio

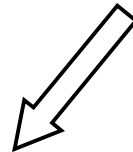


Codice Javascript

```
<html>
<head>
  <title>Esercizio sugli array</title>
</head>
<body>
<script>
<!-- Inizio script JavaScript
var N;
var i;

N = window.prompt("Inserisci la lunghezza dell'array");
N = N * 1;
var vettore = new Array(N);
i = 0;
while (i < N)
{
  if (i%2==0)
  {
    vettore[i] = 10;
  }
  else
  {
    vettore[i] = 20;
  }
  i = i + 1;
}
i = 0;
while (i < N)
{
  window.alert("vettore[" + i + "]=" + vettore[i]);
  i = i + 1;
}
// Fine script -->
</script>
</body>
</html>
```

Inserito per mostrare
il contenuto dell'array

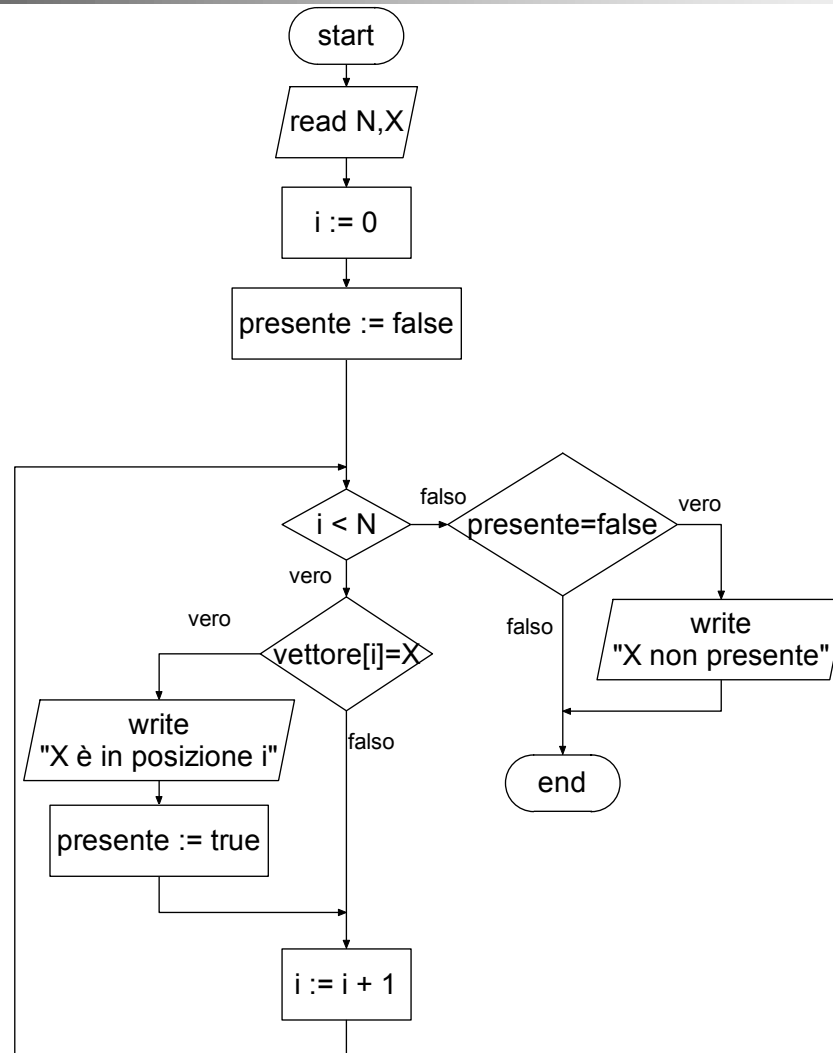




Problema: ricerca di un elemento

- Ricerca di un numero all'interno di un vettore. Assumere che la dimensione del vettore sia uguale ad N e che la prima posizione del vettore sia uguale a 0.
- Esempio: Se $N=4$ e il contenuto del vettore è
 - `vettore[0]=7`
 - `vettore[1]=10`
 - `vettore[2]=555`
 - `vettore[3]=14`
- se proviamo a cercare il numero 555 il risultato deve essere il seguente: "il numero 555 è alla posizione 2 del vettore"
- se proviamo a cercare il numero 90 il risultato deve essere il seguente: "il numero 90 non è presente nel vettore"

Problema: ricerca di un elemento



Codice Javascript: prima parte

```
<html>
<head>
  <title>Esercizio sugli array</title>
</head>
<body>
<script>
<!-- Inizio script JavaScript
var N,X,i,presente;

N = window.prompt("Inserisci la lunghezza dell'array");
N = N * 1;
var vettore = new Array(N);
i = 0;
while (i < N)
{
  vettore[i] = window.prompt("Inserisci vettore[" + i + "]");
  vettore[i] *= 1; //è la stessa cosa di: vettore[i]=vettore[i]*1;
  i = i + 1;
}
X = window.prompt("Inserisci il numero da ricercare");
X = X * 1;
.....
.....
.....
```

Non presente nell'algoritmo
ma necessario





Codice Javascript: seconda parte

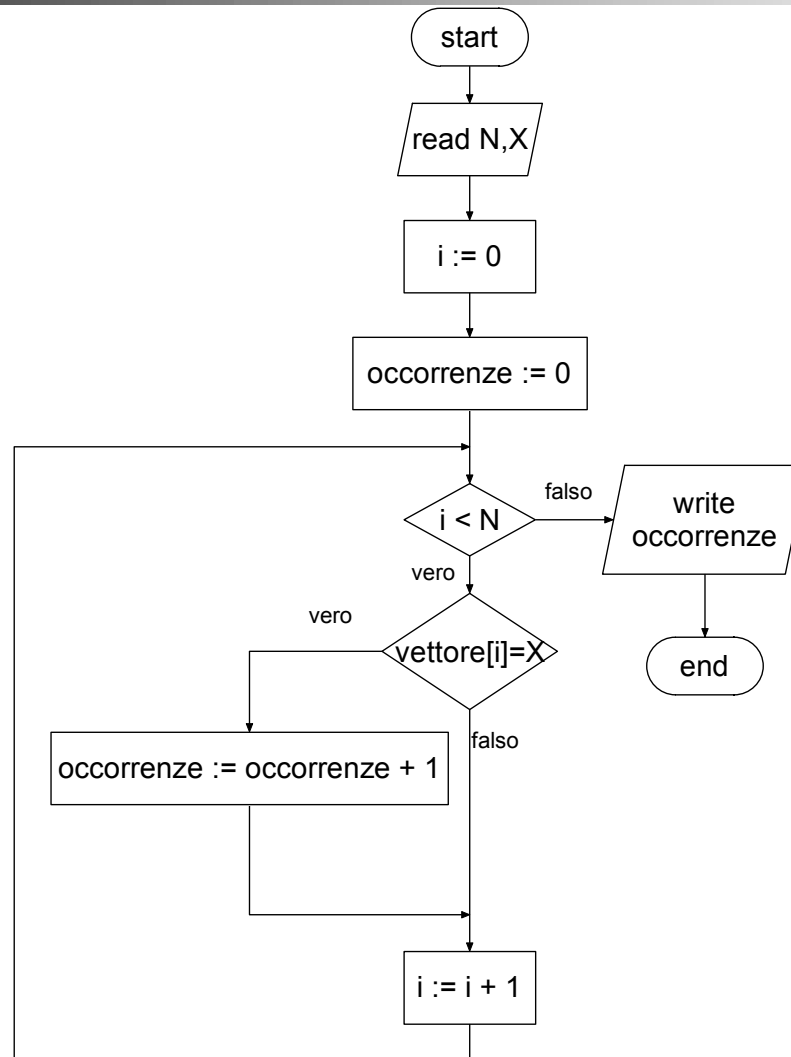
```
.....  
.....  
.....  
i = 0;  
presente = false;  
while (i < N)  
{  
  if(vettore[i]==X)  
  {  
    window.alert("Il numero " + X + " è presente in posizione " + i);  
    presente = true;  
  }  
  i = i + 1;  
}  
if(presente==false)  
{  
  window.alert("Il numero " + X + " non è presente");  
}  
// Fine script -->  
</script>  
</body>  
</html>
```



Problema: conteggio occorrenze

- Conteggio delle occorrenze di un numero all'interno di un vettore. Assumere che la dimensione del vettore sia uguale ad N e che la prima posizione del vettore sia uguale a 0.
- Esempio: Se $N=4$ e il contenuto del vettore è
 - `vettore[0]=555`
 - `vettore[1]=10`
 - `vettore[2]=555`
 - `vettore[3]=14`
- se proviamo a contare quante volte il numero 555 compare nel vettore allora il risultato deve essere il seguente: "il numero 555 compare 2 volte"
- se proviamo a cercare il numero 90 il risultato deve essere il seguente: "il numero 90 compare 0 volte"

Problema: conteggio occorrenze



Codice Javascript: prima parte

```
<html>
<head>
  <title>conteggio occorrenze in array</title>
</head>
<body>
<script>
<!-- Inizio script JavaScript
var N;
var X;
var i;
var occorrenze;

N = window.prompt("Inserisci la lunghezza dell'array");
N = N * 1;
var vettore = new Array(N);
i = 0;
while (i < N)
{
  vettore[i] = window.prompt("Inserisci vettore[" + i + "]");
  vettore[i] *= 1;
  i = i + 1;
}
X = window.prompt("Inserisci il numero da ricercare");
X = X * 1;
.....
.....
```

Non presente nell'algoritmo
ma necessario





Codice Javascript: seconda parte

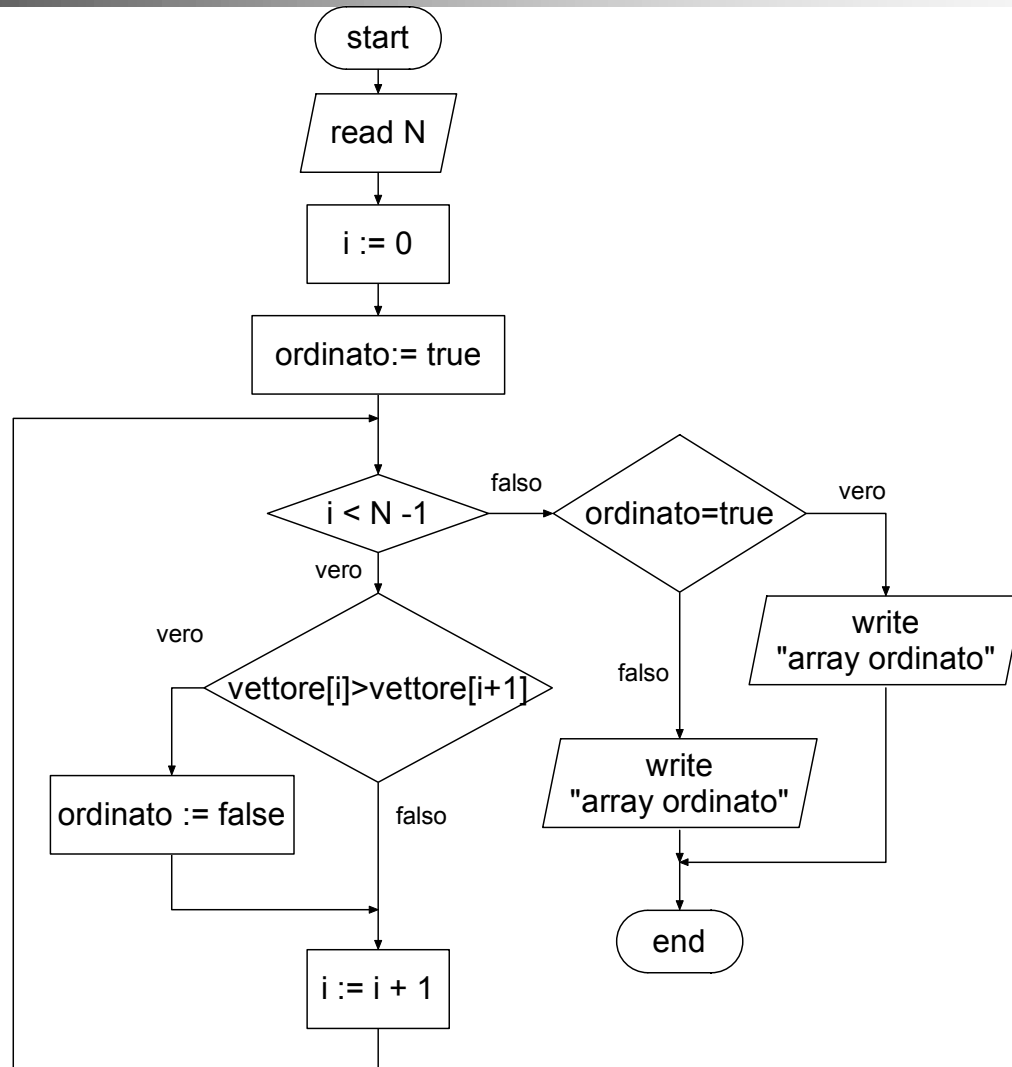
```
.....  
.....  
i = 0;  
occorrenze = 0;  
while (i < N)  
{  
  if(vettore[i]==X)  
  {  
    occorrenze = occorrenze + 1;  
  }  
  i = i + 1;  
}  
window.alert("Il numero " + X + " compare " + occorrenze + " volte nell'array");  
  
// Fine script -->  
  
</script>  
</body>  
</html>
```



Problema: verifica ordinamento

- Scrivere un algoritmo ed un relativo programma in Javascript che calcoli il valore di una variabile booleana che deve essere *true* se l'array è ordinato in maniera crescente e *false* altrimenti e stampi il relativo messaggio. Assumere che la dimensione del vettore sia uguale ad N e che la prima posizione del vettore sia uguale a 0.
- Esempio: Se N=4 e il contenuto del vettore è
 - vettore[0]=555, vettore[1]=10, vettore[2]=555, vettore[3]=14allora la variabile deve valere *false*
- Esempio: Se N=4 e il contenuto del vettore è
 - vettore[0]=12, vettore[1]=17, vettore[2]=555, vettore[3]=1312allora la variabile deve valere *true*

Problema: verifica ordinamento





Codice Javascript: prima parte

```
<html>
<head>
  <title>verifica ordinamento array</title>
</head>
<body>
<script>

<!-- Inizio script JavaScript
var N;
var i;
var ordinato;

N = window.prompt("Inserisci la lunghezza dell'array");
N = N * 1;

var vettore = new Array(N);

i = 0;
while (i < N)
{
  vettore[i] = window.prompt("Inserisci vettore[" + i + "]");
  vettore[i] *= 1;
  i = i + 1;
}
.....
.....
```



Codice Javascript: prima parte

```
.....  
.....  
i = 0;  
ordinato = true;  
while (i < N)  
{  
    if(vettore[i]>vettore[i+1])  
    {  
        ordinato = false;  
    }  
    i = i + 1;  
}  
if(ordinato==true)  
{  
    window.alert("array ordinato");  
}  
else  
{  
    window.alert("array non ordinato");  
}  
  
// Fine script -->  
  
</script>  
</body>  
</html>
```