

ESERCIZIO: RICERCA BINARIA

Soluzione Iterativa

```
int ricerca_bin (int vet[], int el)
{int first=0, last=N-1, med=(first+last)/2;
  int T=0;
  while ((first<=last)&&(T==0))
    { if (el==vet[med])
        T=1;
      else
        if (el < vet[med]) last=med-1;
        else first=med+1;
        med = (first + last) / 2;
    }
  return T;
}
```

1

Ricerca binaria ricorsiva

- Parametri in ingresso:
 - Array in cui cercare
 - Indice first da cui partire
 - Indice last a cui fermarsi nella ricerca
 - Elemento da cercare

- Valori in uscita:
 - **Successo della ricerca**
 - **Posizione dell'elemento nell'array**
 - I due valori sono "condensabili"?
 - La posizione in un array è sempre maggiore o uguale a zero
 - Un numero negativo può essere considerato un insuccesso nella ricerca...

2

RICERCA BINARIA RICORSIVA

```
int ricerca_bin (int vet[], int first, int last, int el) {

    int med;

    if (first > last)
        return -1;
    else {
        med = (first + last) / 2;
        if (el == vet[med])
            return med;
        else
            if (el > vet[med])
                return ricerca_bin(vet, med+1, last, el);
            else
                return ricerca_bin(vet, first, med-1, el);
    }
}
```

RICERCA BINARIA RICORSIVA

Versione "compatta"

```
int binarySearch(int vet[], int dim, int el) {
    int startPos;
    int med = dim / 2;

    if (vet[med] == el)
        return med;
    if (med == 0)
        return -1;
    if (el < vet[med])
    {
        return binarySearch(vet, med, el);
    }
    else {
        startPos = med + 1;
        return startPos +
            binarySearch(&vet[startPos], dim - startPos, el);
    }
}
```

Ricerca binaria: note

- `&vet[startPos]`
 - Indirizzo dell'elemento di posizione `startPos`
 - Sotto-array parzialmente sovrapposto all'array di partenza (`vet`) i cui elementi sono quelli compresi fra `startPos` (compreso) e la fine dell'array
- `startPos + binarySearch(&vet[startPos],`
`dim - startPos, el);`
 - La ricerca riparte dal sotto-array che inizia da `startPos`
 - occorre sommare la posizione di partenza al risultato della sottoricerca
 - la dimensione del sotto-array è `dim - startPos`
 - l'elemento da cercare è sempre lo stesso...