

Nome:

Cognome:

Corso:

## Algoritmica

Compitino 4/4/2002

### Esercizio 1

[punti 12]

Calcolare la complessità in tempo al caso pessimo del seguente algoritmo:

```
Mistero(n)
  if (n < 4) return (1);
  x = Mistero( $\lfloor n/3 \rfloor$ );
  i=1;
  while (i < n) {
    j = 1;
    while (j < n)
      j = 5 * j;
    i = i + 4;
  }
  y = Mistero( $\lfloor n/3 \rfloor$ );
  return(x+y);
```

**Nome:**

**Cognome:**

**Corso:**

## **Algoritmica**

### **Compitino 4/4/2002**

#### **Esercizio 2**

**[punti 18]**

Progettare un algoritmo *divide et impera* per la ricerca di un elemento  $k$  in vettore ordinato  $A$  di  $n$  elementi (si assuma  $n = 3^k$ ), che divida  $A$  in tre parti uguali e organizzi ricorsivamente la ricerca.

- 1) Scrivere lo pseudocodice dell'algoritmo.
- 2) Indicare la relazione di ricorrenza associata alla sua complessità in tempo.
- 3) Risolvere la relazione al punto 2.

Nome:

Cognome:

Corso:

## Algoritmica

### Compitino 4/4/2002

#### Esercizio 1

[punti 12]

Calcolare la complessità in tempo al caso pessimo del seguente algoritmo:

```
Mistero(n)
  if (n < 10) return (1);
  x = Mistero( $\lfloor n/4 \rfloor$ ) + Mistero( $\lfloor n/4 \rfloor$ );
  i=1;
  while (i < n) {
    j = 1;
    while (j < n)
      j = j + 5;
    i = 3 * i;
  }
  y = Mistero( $\lfloor n/4 \rfloor$ );
  return(x+y);
```

**Nome:**

**Cognome:**

**Corso:**

## **Algoritmica**

### **Compitino 4/4/2002**

#### **Esercizio 2**

**[punti 18]**

Dato un vettore ordinato  $A$  contenente  $n$  interi positivi distinti, si deve stabilire se esiste un indice  $i$  tale che  $A[i]=i$ .

- 4) Progettare un algoritmo di complessità in tempo  $O(n)$ , fornendo lo pseudocodice.
- 5) Progettare un algoritmo più efficiente del precedente, fornendo lo pseudocodice e calcolando la sua complessità in tempo.