

## 1. Record di attivazione

Scrivi una funzione in C che dato in ingresso un array di int restituisce il numero maggiore. Scrivi tre versioni: una non ricorsiva, una ricorsiva senza tail recursion e una ricorsiva con tail recursion.

Specifica esattamente i parametri che passi alla procedura, il tipo di passaggio utilizzato e il loro significato. Descrivi anche le assunzioni che fai (ad esempio zero terminated o cose simili).

Scrivi anche un main di esempio in cui chiami le funzioni con un array di tua scelta con almeno 5 elementi. L'array deve essere dichiarato come variabile globale. Non usare altre variabili globali. Nel main deve essere dichiarata la variabile locale **MAX** a cui assegnare il risultato della funzione di cui sopra.

Disegna il record di attivazione per tutte e tre le versioni fino alla massima estensione del record di attivazione. Nel caso di tail recursion, spiega quali ottimizzazioni hai adottato o potresti adottare.

## 2. Cyclone

Riscrivi uno dei metodi implementati nell'esercizio 1 usando Cyclone. Valuta diverse alternative nel passaggio dei parametri.

## 3. Riferimenti in C++

Considera il seguente frammento di codice. Cosa stampa a video?

```
int a = 9;
int & aref = a;
a++;
cout << a << " " << aref << "\n";
int b = 15;
aref = b;
cout << a << " " << aref << "\n";
```

## 4. C++

Considera il seguente codice?

```
class Pet {
public:
    int eat() { return 2; }
    int speak() { return 2; }
    virtual float sleep() { return 3.0; }
};

class Goldfish : private Pet {
public:
    int eat() { return 5; }
    float sleep() { return 7.0; }
};

int main() {
    Goldfish bob;
    cout << bob.eat() << endl;
    cout << bob.speak() << endl;
    cout << bob.sleep() << endl;
    Pet* p = &bob;
    cout << p->sleep() << endl;
}
```

```
}
```

Quale è l'output/effetto prodotto da ogni linea del main? Se contiene un errore scrivi errore, spiega l'errore e ignora la linea.

## 5. Dynamic Binding in Java

Date le seguenti dichiarazioni:

```
class Item{  
  
    int a;  
  
    public Item(int d){a = d; }  
  
    public boolean equals( Item t1){  
        return this.a == t1.a;  
    }  
  
}
```

---

```
Object o1 = new Item(10);  
  
Item a1 = new Item(10);  
  
Item a2 = new Item(10);
```

Quale è il valore delle seguenti tre espressioni spiegando bene (cioè anche il processo di early e late binding dove necessario) il perché (anche se le ritieni errate):

```
o1.equals(a1);  
  
a1.equals(a2);  
  
a2.equals(o1);
```

## 6. Java

Spiega la sottotipazione per i tipi arrays e il problema della covarianza. Fai un esempio in cui dimostri il problema della covarianza per i sottotipi arrays – spiega con esattezza ogni istruzione.

## 7. Semantica assiomatica

Considera il seguente programma che dovrebbe calcolare in d la somma di x e y:

```
d = x;  
while (y != 0) {  
    d++;  
    y--;  
}
```

Come scriveresti le precondizioni e postcondizioni? Come dimostreresti la sua correttezza?