

Matricola_____ Cognome_____ Nome_____ e-mail_____

Fondamenti di Informatica

Prova d'esame del 30/06/2016

Regole d'esame:

1. È vietato parlare con altri studenti
2. È vietato consultare appunti, dispense, libri, in qualunque formato
3. È vietato tenere accesi i telefoni cellulari

Esercizio 1

Si consideri il seguente metodo che riceve in input un array **a**. Indicare, spiegandola brevemente, la complessità asintotica di caso peggiore in funzione della lunghezza **n** dell'array.

```
public static void esercizio(int[] a){
    int n=a.length;
    for(int i=n-1;i>=0;i--){
        int j=i;
        while(j<n){
            j++;
        }
    }
}
```

Esercizio 2

Scrivere un metodo di classe ricorsivo che riceve in input una stringa **s** ed altri parametri se necessario, e restituisce il carattere più "alto" nell'ordinamento lessicografico. Ad esempio, se la stringa fosse "AcWbL", il carattere restituito sarebbe 'c' essendo esso il carattere che segue tutti gli altri nell'ordinamento lessicografico. Scrivere inoltre un metodo per l'avvio della ricorsione.

Note: (1) Una soluzione non ricorsiva non sarà considerata corretta (2) Il metodo non deve utilizzare variabili che non siano locali

Esercizio 3

Il seguente codice contiene cinque errori. Indicarli spiegando brevemente in che cosa consiste l'errore.

```
public interface MyInterface{
    private int x;
    public MyInterface(int x);
    public int metA();
    public double MetB();
    private void metC();
}

public class MyClass implements MyInterface{
    private int z;
    public MyClass(int z){
        this.z=z;
    }
}
```

```

        public int metA(){
            return z*2;
        }
    }

    public class MyExtendedClass extends MyClass{
        private double f;

        public MyExtendedClass (int z, double f){
            this.f=f;
            super(z);
        }
        public double metD(){
            return f/2;
        }
    }
}

```

Esercizio 4

Con riferimento all'algoritmo di ordinamento MergeSort si mostri l'esecuzione del metodo **merge** (che realizza la fusione di due sottosequenze ordinate dell'array) assumendo che esso sia invocato nel seguente modo **merge(dati, temp, 0, 5, 11)** e che **dati** sia il seguente array:

2	5	9	13	24	25	6	8	14	15	20	22
---	---	---	----	----	----	---	---	----	----	----	----

Per rispondere alla domanda si deve mostrare, per ogni passo, quali sono gli elementi (o l'elemento) considerati e come viene modificato l'array temp.

Esercizio 5

Scrivere ciascuno dei numeri indicati nel formato indicato:

- 623 (base 2)
- -324 (complemento a 2 con 10 bit)
- 892 (base 16)