

Matricola_____ Cognome_____ Nome_____ e-mail_____

Fondamenti di Informatica

Prova d'esame del 14/7/2015

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 due interi **n**, ed **m**. Indicare, spiegandola brevemente, la complessità asintotica di caso peggiore in funzione di **n** ed **m**. Spiegare anche a quale “configurazione” di **n** e **m** corrisponde il caso peggiore.

```
public static void esercizio(int n, int m){
    int j=m;
    int c=2*n;

    while(j>n){
        c++;
        j--;
    }

    for(int i=0;i<c;i++){
        System.out.println("i: "+i);
    }
}
```

Esercizio 2

Scrivere un metodo di classe ricorsivo che riceve in input una stringa **s** ed altri parametri se necessario, e restituisce **true** se il numero di caratteri di **s** uguali a 'z' è pari. 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

Siano date le seguenti classi

```
public class Rettangolo{

    private double base;
    private double altezza;

    public Rettangolo(double b, double a){
        base=b;
        altezza=a;
    }

    public double base(){return base;}
}
```

Questo compito è stato discusso e definito collegialmente dalla commissione d'esame di Fondamenti di Informatica

```

    public double altezza(){return altezza;}

    public String toString(){
        String s="Rettangolo";
        s+="base: "+base+" altezza: "+altezza;
        return s;
    }
}

public class Quadrato extends Rettangolo {

    public Quadrato(double lato){
        super(lato,lato);
    }

    public double perimetro(){return 2*(base()+altezza());}

    public String toString(){
        String s="Quadrato";
        s+= " lato: "+base();
        return s;
    }
}

```

Si consideri adesso la seguente classe **ProvaRettangolo**

```

public class ProvaRettangolo{

    public static String metodoA(Rettangolo r){
        return r.toString();
    }

    public static String metodoB(Quadrato q){
        return metodoA(q);
    }

    public static void main(String args[]){

        Quadrato a=new Quadrato(2.0);
        Rettangolo b=a;
        System.out.println(metodoA(a));
        System.out.println(metodoB(b));
        System.out.println(metodoA(b));
        System.out.println(metodoB(a));
        System.out.println("Perimetro "+b.perimetro());

    }
}

```

Indicare se ci sono errori nel main. Se sì indicare quali sono e spiegare brevemente perché sono errori. Infine, per le istruzioni di stampa che non contengono errori dire che cosa viene stampato.

Esercizio 4

Con riferimento all'algoritmo QuickSort, mostrare l'esecuzione della invocazione **partition(dati,0,10)** essendo **dati** il seguente array di interi:

20	3	18	7	27	1	2	30	5	17	16
----	---	----	---	----	---	---	----	---	----	----

PER RISPONDERE ALLA DOMANDA SI DEVE MOSTRARE IL VALORE ASSUNTO DI VOLTA IN VOLTA DALLE VARIABILI UTILIZZATE DAL METODO PARTITION E IL CONTENUTO DELL'ARRAY OGNI VOLTA CHE AVVIENE UNA MODIFICA

Esercizio 5

Dire cosa stampa il seguente codice.

```
class Esercizio5{
    public static void main(String args[]){
        char[] a={'I','T','A','L','I','A'};
        int n=a.length;
        int i=0;

        while(i<n){
            System.out.print(a[i/2*2]);
            i++;
        }
    }
}
```