

# Laboratorio di Programmazione

## Laurea in Bioinformatica

29 novembre 2007

### 1 Programmazione strutturata

#### 1.1 Esercizio 1

Si scriva lo pseudocodice di un programma che calcola la media di due numeri reali. Si ripeta l'esercizio nel caso di due numeri interi.

#### 1.2 Esercizio 2

Si scriva lo pseudocodice di un programma che calcola la media di  $n$  valori  $x_1, \dots, x_n$ , in cui  $n$  è inserito dall'utente in fase di esecuzione del programma. Si ricordi che la media  $m$  di  $n$  valori  $x_1, \dots, x_n$  si calcola mediante la formula

$$m = \frac{1}{n} \sum_{i=1}^n x_i$$

Soluzione:

```
int n,i;
real m,x;
leggi n;
i = 0;
m = 0;
QUANDO i<n
    leggi x;
    m = m+x;
    i = i+1;
RIPETI
m = m/n;
scrivi m;
```

### 1.3 Esercizio 3

Si scriva lo pseudocodice per calcolare il valore massimo di una sequenza di 10 numeri interi inseriti dall'utente in fase di esecuzione.

Soluzione 1:

```
int max,x,i;
max = MINIMO_INT;
i = 0;
QUANDO i<10
    leggi x;
    SE max < x;
        ALLORA
            max = x;
    FINESE
    i = i+1;
RIPETI
scrivi max;
```

N.B.: MINIMO\_INT e' l'intero piu' piccolo rappresentabile con il n. di bit a disposizione. Ad es., se gli interi sono rappresentati con 32 bit, MINIMO\_INT vale  $-2147483648$ .

Soluzione 2:

```
int max,x,i;
leggi x;
max = x;
i = 1;
QUANDO i<10
    leggi x;
    SE max < x;
        ALLORA
            max = x;
    FINESE
    i = i+1;
RIPETI
scrivi max;
```

## 1.4 Esercizio 4

Si scriva lo pseudocodice per calcolare la differenza massima tra  $n$  numeri interi, con  $n$  inserito dall'utente in fase di esecuzione.

Soluzione:

```
int max,min,x,i,n;
max = MINIMO_INT;
min = MASSIMO_INT;
leggi n;
i = 0;
QUANDO i<n
    leggi x;
    SE max < x;
        ALLORA
            max = x;
    FINESE
    SE min > x;
        ALLORA
            min = x;
    FINESE
    i = i+1;
RIPETI
scrivi max-min;
```

## 2 Rappresentazioni numeriche

### 2.1 Esercizio 1

Si scriva lo pseudocodice di un programma che converte un numero intero dalla rappresentazione in una base B generica a quella decimale.

Soluzione:

```
int x, B, i, n, dec;
leggi B; //base
leggi n; //numero di cifre
dec = 0;
i = 0;
QUANDO i < n
    leggi x; //(i+1)-ma cifra, partendo da quella meno significativa
    dec = dec + x * B ^ i
    i = i + 1;
RIPETI
scrivi dec;
```

### 2.2 Esercizio 2

Si scriva lo pseudocodice di un programma che converte un numero intero dalla rappresentazione decimale a quella in una base B generica.

Soluzione:

```
int x, B, dec, i;
leggi dec;
leggi B; //nuova base
i = 0;
QUANDO dec != 0
    x = dec % B;
    scrivi x; //(i+1)-ma cifra, a partire da quella meno significativa
    dec = dec / B; //divisione fra interi!
    i = i + 1;
RIPETI
```

## 2.3 Esercizio 3

Si scriva lo pseudocodice di un programma che converte un numero frazionario dalla rappresentazione decimale a quella in una base B generica.

Soluzione:

```
int x, B, i, m;
float dec;
leggi dec; //numero <1
leggi B; //nuova base
leggi m; //n. max cifre (precisione)
scrivi "0.";
i = 0;
QUANDO ( i < m )
    SE dec != 0
        x = dec * B; //N.B.: moltiplicazione con troncamento del ris.
        scrivi x; //(i+1)-ma cifra, a partire da quella piu' significativa
        dec = dec * B - x;
    FINESE
    i = i + 1;
RIPETI
```

## 2.4 Esercizio 4

Si scriva lo pseudocodice di un programma che esegue la somma di due numeri interi positivi rappresentati in base B.

Soluzione:

```
int ris, x1, x2, B, n, i, overflow, cifra;
overflow = 0;
leggi n; //numero di cifre degli addendi
leggi B; // base
i = 0;
QUANDO i < n
    leggi x1; //(i+1)-ma cifra I addendo, partendo da quella - significativa
    leggi x2; //(i+1)-ma cifra II addendo, partendo da quella - significativa
    cifra = (x1 + x2 + overflow) % B; //(i+1)-ma cifra della somma
    scrivi cifra;
    overflow = (x1 + x2 + overflow) / B; //divisione con troncamento
    i = i + 1;
RIPETI
SE overflow > 0
    cifra = overflow;
    scrivi cifra;
FINESE
```

## 3 JAVA: esercizi introduttivi

### 3.1 Esercizio 1

Si scriva il codice JAVA di un programma che quando eseguito visualizza il messaggio:

Ciao.  
Sono un programma scritto in Java.

Soluzione:

```
import java.io.*;

class Saluto{
    public static void main(String[] args){
        System.out.print("Ciao.\n");
        System.out.print("Sono un programma scritto in JAVA.\n");
    }
}
```

### 3.2 Esercizio 2

Si scriva il codice JAVA di un programma che calcola la media di due numeri interi. Si verifichi il funzionamento con le coppie di valori (3,7) e (4,7) e si cerchi una spiegazione per eventuali errori nel calcolo.

Soluzione:

```
import java.io.*;

class mediaReali{
    public static void main(String[] args) {

        int x;
        int y;
        int media;
        x = 3;
        y = 7;

        media = (x+y)/2;

        System.out.println("Programma per il calcolo della media");
        System.out.print("x = ");
        System.out.println(x);
        System.out.print("y = ");
        System.out.println(y);
        System.out.print("media = ");
        System.out.println(media);
    }
}
```

### 3.3 Esercizio 3

Si scriva il codice JAVA di un programma areaTriangolo che calcola l'area di un triangolo di base 5 e altezza 10. Si ripeta l'esercizio nel caso di un triangolo di base 5 e altezza 7. Il tipo intero e' sufficiente in questo caso per rappresentare l'area?

Soluzione:

```

import java.io.*;

class areaTriangolo{
    public static void main(String[] args) {

        int baseTriangolo;
        int altezzaTriangolo;
        int areaTriangolo;
        baseTriangolo = 5;
        altezzaTriangolo = 10;

        areaTriangolo = (baseTriangolo*altezzaTriangolo)/2;

        System.out.println("Programma per il calcolo dell'area del triangolo");
        System.out.print("Base = ");
        System.out.println(baseTriangolo);
        System.out.print("Altezza = ");
        System.out.println(altezzaTriangolo);
        System.out.print("Area = ");
        System.out.println(areaTriangolo);
    }
}

```

### 3.4 Esercizio 4

Si modifichino gli esercizi 2 e 3 in modo che l'utente possa inserire in fase di esecuzione i valori (x,y) e (base,altezza).

Soluzione:

```

import prog.io.*;

class areaTriangolo{
    public static void main(String[] args) {

        ConsoleInputManager tastiera = new ConsoleInputManager();
        int baseTriangolo;
        int altezzaTriangolo;
        int areaTriangolo;

        System.out.print("Inserisci la base del triangolo: ");
        baseTriangolo = tastiera.readInt();
        System.out.print("Inserisci l'altezza del triangolo: ");
        altezzaTriangolo = tastiera.readInt();

        areaTriangolo = (baseTriangolo*altezzaTriangolo)/2; //att: troncamento ad intero

        System.out.println("Programma per il calcolo dell'area del triangolo");
        System.out.print("Base = ");
        System.out.println(baseTriangolo);
        System.out.print("Altezza = ");
        System.out.println(altezzaTriangolo);
        System.out.print("Area = ");
        System.out.println(areaTriangolo);
    }
}

```



## 4 JAVA: esercizi introduttivi sulla classe String

### 4.1 Esercizio 2

Si scriva un programma che, dopo aver chiesto all'utente di inserire in tre passi successivi il giorno, il mese e l'anno, visualizza la data nel formato gg/mm/aaaa.

Soluzione:

```
import prog.io.*;

class Data{
    public static void main(String[] args) {

        ConsoleInputManager tastiera = new ConsoleInputManager();
        String gg,mm,aaaa;

        System.out.print("Inserisci il giorno (gg): ");
        gg = tastiera.readLine();
        System.out.print("Inserisci il mese (mm): ");
        mm = tastiera.readLine();
        System.out.print("Inserisci l'anno (aaaa): ");
        aaaa = tastiera.readLine();

        System.out.print("La data immessa: "+gg+"/"+mm+"/"+aaaa);
    }
}
```

### 4.2 Esercizio 3

Si scriva un programma che, dopo aver chiesto all'utente di inserire una stringa e due indici (numeri interi), stampa la sottostringa delimitata dai due indici e la sua lunghezza. Soluzione:

```
import prog.io.*;

class Sottostringa{
    public static void main(String[] args) {

        ConsoleInputManager tastiera = new ConsoleInputManager();
        String stringa;
        int i1,i2;

        System.out.print("Inserisci una stringa: ");
        stringa = tastiera.readLine();
        System.out.print("Inserisci l'indice di inizio: ");
        i1 = tastiera.readInt();
        System.out.print("Inserisci l'indice di fine: ");
        i2 = tastiera.readInt();
        System.out.println("La sottostringa richiesta e': "+stringa.substring(i1,i2));
        System.out.println("La lunghezza della sottostringa e': "+stringa.substring(i1,i2).length());
    }
}
```

### 4.3 Esercizio 4

Si scriva un programma che, dopo aver chiesto all'utente di inserire due stringhe di cui la seconda e' una sottostringa della prima, stampa le lunghezze delle due stringhe e l'indice che individua la posizione della sottostringa nella prima stringa.

Soluzione:

```
import prog.io.*;
```

```

class Sottostringa2{
    public static void main(String[] args) {

        ConsoleInputManager tastiera = new ConsoleInputManager();
        String stringa,substringa;

        System.out.print("Inserisci una stringa: ");
        stringa = tastiera.readLine();
        System.out.print("Inserisci una sottostringa della stringa precedente: ");
        substringa = tastiera.readLine();

        System.out.println("La lunghezza della stringa e': "+stringa.length());
        System.out.println("La lunghezza della sottostringa e': "+substringa.length());
        System.out.println("La sottostringa inizia alla posizione: "+stringa.indexOf(substringa));
    }
}

```

## 4.4 Esercizio 5

Si scriva un programma che, dopo aver chiesto all'utente di inserire una stringa ed un carattere, stampa l'indice della prima occorrenza di quel carattere nella stringa e stampa una nuova stringa nella quale quel carattere e' sostituito dal carattere "X".

```

import prog.io.*;

class CharToX{
    public static void main(String[] args) {

        ConsoleInputManager tastiera = new ConsoleInputManager();
        String stringa;
        String carattere;
        int posizione;

        System.out.print("Inserisci una stringa: ");
        stringa = tastiera.readLine();
        System.out.print("Inserisci un carattere: ");
        carattere = tastiera.readLine();

        posizione = stringa.indexOf(carattere);
        System.out.println("La nuova stringa e': "+stringa.substring(0,posizione)+
                           "X"+stringa.substring(posizione+1,stringa.length()));
    }
}

```

## 5 JAVA: i costrutti di selezione "if/else" e iterazione "while"

### 5.1 Esercizio 1

Scrivere un programma che chiede all'utente di inserire i valori di due variabili intere var1 e var2 e, in base ad opportuni confronti fra le variabili, stampa la frase "Il valore di var1 è [uguale al | maggiore del | minore del] valore di var2".

### 5.2 Esercizio 2

Sul prezzo di un prodotto viene praticato lo sconto del 3% se costa meno di 500 euro e del 5% per prezzi superiori a 500 euro. Scrivere un programma che calcoli il prezzo da pagare.

### 5.3 Esercizio 3

Scrivere un programma che, lette due stringhe inserite dall'utente, indichi se sono uguali o diverse. Nel caso di stringhe diverse l'applicazione dovrà visualizzarle in ordine lessicografico e poi in ordine di lunghezza [es. 3.6 pag 80].

```
import prog.io.*;

class confrontoStringhe{
    public static void main(String[] args){

        ConsoleInputManager in = new ConsoleInputManager();
        String st1, st2;
        int lenst1, lenst2;

        System.out.println("Confronto di stringhe ");
        System.out.print("Inserire la prima stringa: ");
        st1 = in.readLine();
        System.out.print("Inserire la seconda stringa: ");
        st2 = in.readLine();

        int cfr = st1.compareTo(st2);
        if(cfr==0)
            System.out.println("Le due stringhe sono uguali");
        else {
            lenst1=st1.length();
            lenst2=st2.length();
            System.out.println("Ordine lessicografico:");
            if(cfr<0) {
                System.out.println(st1);
                System.out.println(st2);}
            else {
                System.out.println(st2);
                System.out.println(st1);
            }
            System.out.println("Ordine di lunghezza:");
            if (lenst1==lenst2)
                System.out.println("Le due stringhe hanno stessa lunghezza");
            else if(lenst1<lenst2){
                System.out.println(st1);
                System.out.println(st2);}
            else {
                System.out.println(st2);
                System.out.println(st1);}
        }
    }
}
```

## 5.4 Esercizio 4

Scrivere un programma per calcolare il valore massimo di una sequenza di numeri interi inseriti dall'utente in fase di esecuzione. Per gestire l'interazione con l'utente si usi una frase del tipo "Vuoi inserire un altro numero (s/n)?".

```
import prog.io.*;

class MaxDiN{
    public static void main(String[] args){

        ConsoleInputManager in = new ConsoleInputManager();
        String ans;
        int num, max;

        System.out.println("Massimo fra N numeri ");

        System.out.print("Inserisci il primo numero: ");
        num = in.readInt();
        max = num;

        System.out.println("Vuoi inserire un nuovo numero (s/n)? ");
        ans = in.readLine();

        while(ans.compareTo("s")==0){
            System.out.print("Inserisci il nuovo numero: ");
            num = in.readInt();
            if(num>max)
                max=num;

            System.out.println("Vuoi inserire un nuovo numero (s/n)? ");
            ans = in.readLine();
        }
        System.out.println("Ecco il massimo: "+max);
    }
}
```

## 5.5 Esercizio 5

Si scriva un programma che, dopo aver chiesto all'utente di inserire una stringa ed un carattere, stampa l'indice di TUTTE le occorrenze di quel carattere nella stringa e stampa una nuova stringa nella quale TUTTE le occorrenze di quel carattere sono sostituite dal carattere "X". Si risolva il problema evitando l'uso del metodo "replace" della classe String.

Soluzione 1:

```
import prog.io.*;

class AllCharToX{
    public static void main(String[] args) {

        ConsoleInputManager tastiera = new ConsoleInputManager();
        String stringa, nuovastringa;
        String carattere;
        int posizione, posizioneprec;

        System.out.print("Inserisci una stringa: ");
        stringa = tastiera.readLine();
        System.out.print("Inserisci un carattere: ");
        carattere = tastiera.readLine();

        posizioneprec=0;
        posizione = stringa.indexOf(carattere);

        System.out.println("posizione: "+posizione);
        nuovastringa="";
    }
}
```

```

while(posizione>=0){
    nuovastringa=nuovastringa+stringa.substring(posizioneprec,(posizione+posizioneprec))+"X";
    posizioneprec=posizioneprec+posizione+1;
    System.out.println("posizioneprec: "+posizioneprec);

    posizione = stringa.substring(posizioneprec,stringa.length()).indexOf(carattere);
    System.out.println("posizione: "+posizione);
}

nuovastringa=nuovastringa+stringa.substring(posizioneprec,stringa.length());
System.out.println("La nuova stringa e': "+nuovastringa);
}
}

```

Soluzione 2:

```

import prog.io.*;

class AllCharToXver2{
    public static void main(String[] args) {

        ConsoleInputManager tastiera = new ConsoleInputManager();
        String stringa;
        String carattere;
        int posizione;

        System.out.print("Inserisci una stringa: ");
        stringa = tastiera.readLine();
        System.out.print("Inserisci un carattere: ");
        carattere = tastiera.readLine();

        posizione = stringa.indexOf(carattere);
        System.out.println("posizione: "+posizione);

        while(posizione>=0){
            stringa=stringa.substring(0,posizione)+"X"+stringa.substring(posizione+1,stringa.length());
            posizione = stringa.indexOf(carattere);

            System.out.println("posizione: "+posizione);
        }
        System.out.println("La nuova stringa e': "+stringa);
    }
}

```

## 6 JAVA: uso di oggetti

### 6.1 Esercizio 1

Adoperando i metodi forniti dalla classe `Frazione` (cfr. pp. 72-73 del libro del corso) si scriva un programma java il quale:

1. Accetta da tastiera due coppie di valori interi positivi, creando due oggetti `Frazione` corrispondenti rispettivamente riferiti dalle variabili `f1` e `f2`
2. Calcola e stampa la differenza tra la frazione più grande e la più piccola e ne salva il riferimento in `f3` (N.B.: il risultato sarà per forza non negativo)
3. Calcola e stampa la divisione tra frazione più piccola e la più grande e ne salva il riferimento in `f4` (N.B.: il risultato sarà per forza minore di uno)
4. Copia l'oggetto riferito da `f4` in un nuovo oggetto `Frazione` riferito da `f5`
5. Verifica se gli ultimi due oggetti sono uguali oppure no e stampa quello riferito da `f5`
6. Verifica se gli ultimi due riferimenti sono uguali oppure no.

```
import prog.io.*;
import prog.utili.Frazione;

class FrazioneEs1 {
    static public void main (String args[]) {
        ConsoleInputManager tastiera = new ConsoleInputManager();
        ConsoleOutputManager video = new ConsoleOutputManager();

        // Accetta da tastiera due coppie di valori interi positivi, creando due oggetti Frazione
        // corrispondenti rispettivamente riferiti dalle variabili f1 e f2
        Frazione f1 = new Frazione(tastiera.readInt("Numeratore Frazione 1: "),tastiera.readInt("Denominatore Frazione 1: "));
        Frazione f2 = new Frazione(tastiera.readInt("Numeratore Frazione 2: "),tastiera.readInt("Denominatore Frazione 2: "));

        // Calcola e stampa la differenza tra la frazione più grande e la più piccola e ne salva
        // il riferimento in f3 (N.B.: il risultato sarà per forza non negativo)
        Frazione f3;
        if (f1.isMaggiore(f2))
            f3 = f1.meno(f2);
        else
            f3 = f2.meno(f1);
        video.println("f3 = " + f3.toString());

        // Calcola e stampa la divisione tra frazione più piccola e la più grande e ne salva il
        // riferimento in f4 (N.B.: il risultato sarà per forza minore di uno)
        Frazione f4;
        if (f1.isMinore(f2))
            f4 = f1.diviso(f2);
        else
            f4 = f2.diviso(f1);
        video.println("f4 = " + f4.toString());

        // Copia l'oggetto riferito da f4 in un nuovo oggetto Frazione riferito da f5
        Frazione f5 = new Frazione(f4.getNumeratore(),f4.getDenominatore());

        // Verifica se gli ultimi due oggetti sono uguali oppure no e stampa quello riferito da f5
        if (f4.equals(f5))
            video.println("Gli oggetti riferiti da f4 e f5 sono uguali.");
        else
            video.println("Gli oggetti riferiti da f4 e f5 sono diversi.");
        video.println("f5 = " + f5.toString());

        // Verifica se gli ultimi due riferimenti sono uguali oppure no
        if (f4==f5)
            video.println("f4 e f5 sono uguali.");
        else
            video.println("f4 e f5 sono diversi.");
    }
}
```

## 6.2 Esercizio 2

Adoperando i metodi forniti dalla classe `Frazione` si scriva un programma java il quale:

1. Accetta da tastiera due valori interi positivi, creando un oggetto `Frazione` riferito dalla variabile `f1`
2. Calcola e stampa il reciproco della frazione appena data e ne salva il riferimento in `f2` (N.B.: qual è il reciproco di una frazione  $a/b$  ?)
3. Calcola e stampa la moltiplicazione tra le frazioni riferite da `f1` e `f2` e ne salva il riferimento in `f3` (N.B.: quanto deve valere questa moltiplicazione ?)
4. Calcola e stampa la divisione tra le frazioni riferite da `f1` e `f2` e ne salva il riferimento in `f4` (N.B.: quanto deve valere questa divisione ?)
5. Calcola il quadrato della frazione riferita da `f1` e ne salva il riferimento in `f5`
6. Verifica se gli oggetti riferiti da `f4` e `f5` sono uguali oppure no e stampa quello riferito da `f5`

```
import prog.io.*;
import prog.utili.Frazione;

class FrazioneEs2 {
    static public void main (String args[]) {

        ConsoleInputManager tastiera = new ConsoleInputManager();
        ConsoleOutputManager video = new ConsoleOutputManager();

        Frazione f1 = new Frazione(tastiera.readInt("Numeratore Frazione 1: "),tastiera.readInt("Denominatore Frazione 1: "));

        Frazione f2 = new Frazione(f1.getDenominatore(),f1.getNumeratore());
        video.println("f2 = " + f2.toString());

        Frazione f3 = f1.per(f2);
        video.println("f3 = " + f3.toString());

        Frazione f4 = f1.diviso(f2);
        video.println("f4 = " + f4.toString());

        Frazione f5 = f1.per(f1);
        video.println("f5 = " + f5.toString());

        if (f4.equals(f5))
            video.println("Gli oggetti riferiti da f4 e f5 sono uguali.");
        else
            video.println("Gli oggetti riferiti da f4 e f5 sono diversi.");
    }
}
```

## 7 JAVA: il costrutto di iterazione "for"

### 7.1 Esercizio 1

Si scriva il programma RuotaStringa che:

1. acquisisce da standard input una stringa
2. produce una nuova stringa, la quale è una versione ruotata della stringa immessa. La rotazione avviene spostando tutti i simboli in base al valore assunto da un indice di rotazione, con la regola che simboli che uscirebbero dalla stringa rientrano dalla parte opposta:
  - un indice positivo n provoca la rotazione a destra di n posizioni della stringa
  - un indice negativo n provoca la rotazione a sinistra di n posizioni della stringaEsempio: n = 1 trasforma "remo" in "orem"  
Esempio: n = -1 trasforma "remo" in "emor"
3. stampa la stringa così ruotata

N.B.: e' obbligatorio l'utilizzo del costrutto "for" per la soluzione!

```
import prog.io.*;

class RuotaStringa {
    public static void main(String[] args) {

        ConsoleInputManager in = new ConsoleInputManager();
        ConsoleOutputManager out = new ConsoleOutputManager();

        String iniziale = in.readLine("Inserisci stringa non vuota da ruotare: ");
        String finale = iniziale.toString();          // equivale a new String(iniziale.toString());

        int rotazione = in.readInt("Inserisci indice di rotazione: ");
        out.println("Rotazione...");

        if (rotazione>0)
            for (int r=0; r<rotazione; r++)
                finale = finale.substring(finale.length()-1) + finale.substring(0,finale.length()-1);
        else
            for (int r=0; r>rotazione; r--)
                finale = finale.substring(1) + finale.substring(0,1);

        out.println("Stringa ruotata: " + finale);
    }
}
```

### 7.2 Esercizio 2

Si scriva il programma TestPalindroma che verifica se la stringa immessa dall'utente è una parola palindroma. Esempi di parole palindrome sono: "ara", "anna", "otto", "ingegni", etc.

```
import prog.io.*;

class TestPalindroma {
    public static void main(String[] args) {

        ConsoleInputManager in = new ConsoleInputManager();
        boolean palindroma=true;
        String stringa = in.readLine();
        //int substsx=0;
        //int substdx=stringa.length()-1;

        for (int sx=0, dx=stringa.length()-1 ; sx<=dx ; sx++, dx--) {
```



```

        if(stringa.charAt(sx) != stringa.charAt(dx)) {
            palindroma=false;
        }
    }
    if(palindroma)
        System.out.println("La stringa \"" + stringa + "\" e' palindroma.");
    else
        System.out.println("La stringa \"" + stringa + "\" non e' palindroma.") ;
}
}

```

## 7.3 Esercizio 3

Si scriva il programma TutteLePalindrome che:

1. acquisisce dal file testuale "stringa.txt" una stringa
2. trova tutte le sottostringhe palindrome contenuta nella stringa d'ingresso  
Es.: in "1211123212221", la sottostringhe palindrome sono "121", "11", "111", "21112", "12321", etc.
3. stampa le sottostringhe trovate

N.B.: la soluzione riportata effettua il test di palindromicit  su sottostringhe di lunghezza  $\geq 2$ .

```

import prog.io.*;

class TutteLePalindrome {
    public static void main(String[] args) {

        FileInputManager inFile = new FileInputManager("stringa.txt");
        ConsoleOutputManager out = new ConsoleOutputManager();

        String stringa = inFile.readLine();

        out.println("Input da file: " + stringa);
        out.println("Ricerca delle sottostringhe palindrome...");

        for (int substsx=0; substsx<stringa.length()-1; substsx++){
            for (int substdx=stringa.length()-1; substdx>substsx; substdx--){

                /***** questo e' il solito check di palindromicita' *****/
                boolean palindroma=true;
                for (int sx=substsx, dx=substdx ; sx<=dx ; sx++, dx--) {
                    if(stringa.charAt(sx) != stringa.charAt(dx)) {
                        palindroma=false;
                    }
                }
                if(palindroma)
                    out.println("Sottostringa palindroma: " + stringa.substring(substsx,substdx+1));
                /*****/
            }
        }
    }
}

```

## 8 JAVA: cast implicito ed esplicito, operatori postfissi e prefissi

### 8.1 Esercizio 1

Si scriva il programma TypeCast1 che:

1. dichiara tre variabili intere i,j, e k con valori rispettivamente 3,4, e 6
2. usando il cast di tipo implicito o esplicito, calcola la media corretta (rappresentata come float) in almeno 3 modi diversi e la media troncata (rappresentata con un intero) in almeno due modi diversi

```
import prog.io.*;

class TypeCast1 {
    public static void main(String[] args) {

        ConsoleInputManager in = new ConsoleInputManager();
        ConsoleOutputManager out = new ConsoleOutputManager();

        int i=3, j=4, k=6;
        float media_f;
        int media_i;

        media_f = (i+j+k)/3.0f;
        out.println("Media corretta: " + media_f);

        media_f = ((float)i+j+k)/3;
        out.println("Media corretta: " + media_f);

        media_f = (float)(i+j+k)/3;
        out.println("Media corretta: " + media_f);

        media_i = (i+j+k)/3;
        out.println("Media troncata: " + media_i);

        media_i = (int)((i+j+k)/3.0);
        out.println("Media troncata: " + media_i);
    }
}
```

## 8.2 Esercizio 2

Si scriva il programma TypeCast2 che:

1. legge da tastiera tre variabili intere i,j, e k
2. assegna la media dei tre numeri a una variabile del tipo strettamente necessario a evitare la perdita di informazioni

[suggerimento: si usi l'operatore % per le valutazioni sul tipo da usare]

```
import prog.io.*;

class TypeCast2 {
    public static void main(String[] args) {

        ConsoleInputManager in = new ConsoleInputManager();
        ConsoleOutputManager out = new ConsoleOutputManager();

        int i, j, k, somma;
        i=in.readInt("Inserisci il valore di i:");
        k=in.readInt("Inserisci il valore di j:");
        j=in.readInt("Inserisci il valore di k:");

        somma=i+j+k;
        if(somma%3 == 0){
            int media=somma/3;
            out.println("Media (int): " + media);
        }
        else {
            double media=somma/3.0;
            out.println("Media (double): " + media);
        }
    }
}
```

## 8.3 Esercizio 3

Si scriva il programma TypeCastRand che:

1. legge da tastiera il valore di un intero N
2. genera N numeri casuali interi compresi nell'intervallo [5,10]
3. ne calcola la media

[suggerimento: si usi la funzione Math.random() che genera numeri casuali di tipo double nell'intervallo [0,1] con distribuzione uniforme ]

```
import prog.io.*;

class TypeCastRand {
    public static void main(String[] args) {

        ConsoleInputManager in = new ConsoleInputManager();
        ConsoleOutputManager out = new ConsoleOutputManager();

        int sampleInt;
        int sumInt=0;
        int min=5, Max=10;
        double media;

        int N=in.readInt("Inserisci il n. di campioni: ");

        for(int i=0; i<N; i++){
            sampleInt=(int)(min+(Max-min)*Math.random());
            if(N<=5) out.println(sampleInt);
        }
    }
}
```

```

        sumInt+=sampleInt;
    }

    media = (double)sumInt/N;
    out.println("Media: "+media);
}
}

```

## 8.4 Esercizio 4

Il programma PrePost\_num sia scritto usando gli operatori pre e postfissi:

1. stampa il risultato di tre operazioni di incremento postfisso in sequenza (i messaggi a video siano del tipo "i=0", "stampa di i++: 0", "stampa di i++: 1", etc);
2. si ripeta la stessa cosa con l'incremento prefisso
3. si valutino le espressioni i=i++ e i=++i e si stampino a video dei messaggi del tipo ("i=0", "stampa di i=i++: ...", etc);
4. si valutino le espressioni i==i++ e i==++i come condizioni del costrutto IF e si stampi un messaggio che dica se sono sempre vere o sempre false

```

import prog.io.*;

class PrePost_num {
    public static void main(String[] args) {

        ConsoleInputManager in = new ConsoleInputManager();
        ConsoleOutputManager out = new ConsoleOutputManager();

        int i=0;
        out.println("i=0");
        out.println("stampa di i++: " + i++);
        out.println("stampa di i++: " + i++);
        out.println("stampa di i: " + i);
        out.println("*****");
        /*****/
        i=0;
        out.println("i=0");
        out.println("stampa di ++i: " + ++i);
        out.println("stampa di ++i: " + ++i);
        out.println("stampa di i: " + i);
        out.println("*****");
        /*****/
        i=0;
        out.println("i=0");
        out.println("stampa di i=i++: " + (i=i++));
        out.println("*****");

        i=0;
        out.println("i=0");
        out.println("stampa di i=++i: " + (i=++i));
        out.println("*****");
        /*****/
        if(i==i++)
            out.println("i==i++ e' sempre vera");
        else
            out.println("i==i++ e' sempre falsa");

        if(i==++i)
            out.println("i==++i e' sempre vera");
        else
            out.println("i==++i e' sempre falsa");
        out.println("*****");
    }
}

```