Laboratorio di Programmazione I (CdL Informatica) &

Laboratorio di Programmazione (CdL Matematica)

a.a. 2004-2005

Prova Intermedia - 4 Novembre 2004

Alcune Soluzioni

Esercizio 1. Si consideri il seguente frammento di codice in Java:

```
String[] a = b;
int i = 0, k = 1;
String s = "";
while (k<a.length) {
  s = a[k].substring(i,k);
  if (s.equals(a[k-1]))
    i++;
  k++;
}
```

Determinare il valore finale delle variabili `i`, `k` ed `s` nei seguenti casi:
1) `b={"ab","bcde","cde","efghh"}`;
2) `b={"a","bb","bbc","fbb","afbbde"}`.

Soluzione
1) `i = 0`, `k = 4`, `s = "efg"`.
2) `i = 2`, `k = 5`, `s = "fbb"`.

Esercizio 2. Si consideri il seguente frammento di codice in Java:

```
int[][] m = a;
int i = m.length-1, j = 0;
boolean p = true;
while (i >= 0 && p) {
  for (j=0; j<m[i].length-1; j++) {
     if (m[i][j] % m[i][j+1] == 0)
       p = false;
  }
  i--;
}
```

Determinare il valore finale delle variabili `i`, `j` e `p` nei seguenti casi:
1) `a={{1,3,2,1},{4,2,6,3},{3,2,7,3}}`;
2) `a={{-9,3},{6,-3},{4,2},{3,2}}`.

Soluzione
1) `i = 0`, `j = 3`, `p = false`.
2) `i = 1`, `j = 1`, `p = false`.

Esercizio 3. Scrivere un metodo che, dati due array di interi a e b, restituisce *true* se tutti gli elementi di a compaiono in b e viceversa, altrimenti il metodo restituisce *false*. Ad esempio, dati gli array {3,3,-4,7} e {7,3,7,-4,-4}, il metodo restituisce *true*.

Soluzione

```
public static boolean multisetEqual (int[] a, int[] b) {
  return (subarray(a,b) && subarray(b,a));
}

public static boolean subarray (int[] a, int[] b) {
  for (int i=0; i<a.length; i++) {
    if (!occorre(a[i],b))
      return false;
  }
  return true;
}

public static boolean occorre (int x, int[] a) {
```

```
  for (int i=0; i<a.length; i++) {
    if (x == a[i]) return true;
  }
  return false;
}
```

Esercizio 4. Scrivere un metodo che, dato un array bidimensionale di interi a, restituisce *true* se la somma degli elementi a_{ij}, la cui somma degli indici i e j è pari, è uguale alla somma degli elementi a_{ij}, la cui somma degli indici i e j è dispari. Altrimenti il metodo restituisce *false*. Ad esempio, l'array {{3,-2,4},{7,-4,1,5},{-2,0}} soddisfa tale proprietà.

Soluzione

```
public static boolean m4 (int[][] a) {
  int sump = 0, sumd = 0;
  for (int i=0; i<a.length; i++) {
    for (int j=0; j<a[i].length; j++) {
      if ((i+j)%2 == 0)
        sump = sump + a[i][j];
      else
        sumd = sumd + a[i][j];
    }
  }
  return (sump == sumd);
}
```

Nelle altre versioni dell'Esercizio 4. è sufficiente apportare le seguenti modifiche:
- se occorre controllare che il prodotto degli indici i e j sia pari, si sostituisce `i+j` con `i*j` nell'espressione del modulo;
- se occorre calcolare il prodotto degli elementi a_{ij}, le variabili locali in cui si accumulano i prodotti devono essere inizializzate con `1`;
- si sostituisce l'operatore `==` nell'espressione booleana `(sump == sumd)` con `>` e `<` rispettivamente.

Esercizio 5. Scrivere un metodo che, dati un array di stringhe a ed un intero k (k>0), restituisce un array di stringhe costituito dalle stringhe di a la cui lunghezza è maggiore di k. Ad esempio, se l'array a è {"haskell","java","pascal","ada"} e k=4, il metodo restituisce l'array {"haskell","pascal"}.
N.B. L'array restituito non deve essere piú lungo del necessario.

Soluzione

```
public static String[] m5 (String[] a, int k) {
  int cont = 0;
  for (int i=0; i<a.length; i++) {
    if (a[i].length() > k)
      cont++;
  }
  String[] b = new String[cont];
  int j = 0;
  for (int i=0; i<a.length; i++) {
    if (a[i].length() > k) {
      b[j] = a[i];
```

```
        j++;
      }
    }
    return b;
  }
```

Nelle altre versioni dell'Esercizio 5. è sufficiente sostituire l'operatore `>` nell'espressione booleana `(a[i].length() > k)` con `<=`, `==` e `<` rispettivamente.

Le altre versioni dell'Esercizio 1. e dell'Esercizio 2.

Esercizio 1. Si consideri il seguente frammento di codice in Java:

```
String[] a = b;
int i = 0, j = 1;
String s = "";
while (j<a.length) {
  s = a[j].substring(i,j);
  if (a[j-1].equals(s))
    i++;
  j++;
}
```

Determinare il valore finale delle variabili `i`, `j` ed `s` nei seguenti casi:
1) `b={"zx","xyzw","yzw","wsttp"}`;
2) `b={"r","qq","qqr","rqq","rrqqvz"}`.

Soluzione
1) `i = 0`, `j = 4`, `s = "wst"`.
2) `i = 2`, `j = 5`, `s = "rqq"`.

Esercizio 2. Si consideri il seguente frammento di codice in Java:

```
int[][] b = a;
int i = 0, j = 0;
boolean p = false;
while (i < b.length && !p) {
  for (j=0; j<b[i].length-1; j++) {
     if (b[i][j+1] % b[i][j] == 0)
       p = true;
  }
  i++;
}
```

Determinare il valore finale delle variabili `i`, `j` e `p` nei seguenti casi:
1) `a={{3,2,5,2},{2,4,3,6},{1,5,7,3}}`;
2) `a={{-3,12},{3,-6},{4,2},{7,2}}`.

Soluzione
1) `i = 2`, `j = 3`, `p = true`.
2) `i = 1`, `j = 1`, `p = true`.

Esercizio 1. Si consideri il seguente frammento di codice in Java:

```
String[] a = b;
int c = 0, k = 1;
String s = "";
while (k<a.length) {
  s = a[k].substring(c,k);
  if (s.equals(a[k-1]))
    c++;
  k++;
}
```

Determinare il valore finale delle variabili `c`, `k` ed `s` nei seguenti casi:
1) `b={"fg","gxfg","xfg","gggxf"}`;
2) `b={"d","aa","aad","daa","aaddkj"}`.

Soluzione
1) `c = 0`, `k = 4`, `s = "ggg"`.
2) `c = 1`, `k = 5`, `s = "add"`.

Esercizio 2. Si consideri il seguente frammento di codice in Java:

```
int[][] b = c;
int i = b.length-1, j = 0;
boolean q = false;
while (i >= 0 && !q) {
  for (j=b[i].length-1; j>0; j--) {
     if (b[i][j-1] % b[i][j] == 0)
       q = true;
  }
  i--;
}
```

Determinare il valore finale delle variabili `i`, `j` e `q` nei seguenti casi:
1) `c={{5,3,2,3},{6,2,9,3},{1,5,7,3}}`;
2) `c={{-8,4},{6,3},{4,2},{7,2}}`.

Soluzione
1) `i = 0`, `j = 0`, `q = true`.
2) `i = 1`, `j = 0`, `q = true`.

Esercizio 1. Si consideri il seguente frammento di codice in Java:

```
String[] a = b;
int r = 0, i = 1;
String s = "";
while (i<a.length) {
  s = a[i].substring(r,i);
  if (a[i-1].equals(s))
    r++;
  i++;
}
```

Determinare il valore finale delle variabili `r`, `i` ed `s` nei seguenti casi:
1) `b={"xz","zyxw","yxw","wwxxy"}`;
2) `b={"h","kk","kkh","hkk","hhkkfg"}`.

Soluzione
1) `r = 0`, `i = 4`, `s = "wwx"`.
2) `r = 2`, `i = 5`, `s = "hkk"`.

Esercizio 2. Si consideri il seguente frammento di codice in Java:

```
int[][] a = m;
int i = 0, j = a[0].length-1;
boolean q = false;
while (i < a.length && !q) {
  for (j=a[i].length-1; j>0; j--) {
     if (a[i][j-1] % a[i][j] == 0)
       q = true;
  }
  i++;
}
```

Determinare il valore finale delle variabili `i`, `j` e `q` nei seguenti casi:
1) `m={{1,5,7,3},{6,2,9,3},{5,3,2,3}}`;
2) `m={{7,2},{5,3},{-8,4},{6,3}}`.

Soluzione
1) `i = 2`, `j = 0`, `q = true`.
2) `i = 3`, `j = 0`, `q = true`.