# Programação 1

## Aula Arrays

Valeri Skliarov, Prof. Catedrático

Email: skl@ua.pt
URL: http://sweet.ua.pt/skl/

Departamento de Eletrónica, Telecomunicações e Informática
Universidade de Aveiro

http://elearning.ua.pt/

# Objetivos da aula 11:

1. *Arrays* multidimensionais.

2. *Arrays* de *Strings*.

3. *Arrays* de registos (objetos definidos através de classes)

# *Arrays* bidimensionais

Uma sequência bidimensional é uma sequência de sequências [][].

Exemplo de uma sequência bidimensional:

```
array1.length = 3
array1[0].length = 2
1    2    3    4    5    6
Press any key to continue . . .
```

```
import java.util.*;
public class ex11 {
            static Scanner read = new Scanner(System.in);
public static void main (String args[]) {
  int array[][] = { {1,2}, {3,4}, {5,6}};
System.out.println("array.length = " + array.length);          // 3
System.out.println("array[0].length = " + array[0].length);   // 2
  for(int i = 0; i<array.length; i++)
    for(int j = 0; j<array[0].length; j++)
      System.out.print(array[i][j] + "   ");   // 1  2   3   4   5   6
  System.out.println();
  }}
```

# *Arrays* bidimensionais e tridimensionais

Uma sequência bidimensional é uma sequência de sequências [][].

Uma sequência tridimensional é uma sequência de sequências de sequências [][][], etc.

Exemplo de uma sequência bidimensional:

```
int Two_D_array[][] = new int[N][M];
System.out.println("Two dimentional array");
for(int i = 0; i < Two_D_array.length; i++)
      for(int j = 0; j < Two_D_array[0].length; j++)
          Two_D_array[i][j] = i+j;
  for(int i = 0; i < Two_D_array.length; i++)   {
      for(int j = 0; j < Two_D_array[0].length; j++)
          System.out.print(Two_D_array[i][j]+ " ");
System.out.println();
```

```
Two dimentional array
0 1 2
1 2 3
Press any key to continue . . .
```

# *Arrays* tridimensionais

Uma sequência tridimensional é uma sequência de sequências de sequências [][][], etc.

Exemplo de uma sequência tridimensional:

```
int Three_D_array[][][] = new int[2][3][4];
 System.out.println("\nThree dimentional array");
 for(int i = 0; i < Three_D_array.length; i++)
    for(int j = 0; j < Three_D_array[0].length; j++)
       for(int k = 0; k < Three_D_array[0][0].length; k++)
          Three_D_array[i][j][k] = i+j+k;
 for(int i = 0; i < Three_D_array.length; i++)   {
    for(int j = 0; j < Three_D_array[0].length; j++)   {
       for(int k = 0; k < Three_D_array[0][0].length; k++)
          System.out.print(Three_D_array[i][j][k]+ " ");
    System.out.println();                              }
 System.out.println();                              }
```

```
Three dimentional array
0 1 2 3
1 2 3 4
2 3 4 5

1 2 3 4
2 3 4 5
3 4 5 6

Press any key to continue . . .
```

# *Arrays* quatrodimensionais

Uma sequência quatroimensional é uma sequência de sequências de sequências … [][][][], etc.

Exemplo de uma sequência tridimensional:

```
 int Four_D_array[][][][] = new int[2][3][4][5];
  System.out.println("\nFour dimentional array");
for(int  a = 0; a < Four_D_array.length; a++)
  for(int i = 0; i < Four_D_array.length[0]; i++)
     for(int j = 0; j < Four_D_array[0][0].length; j++)
        for(int k = 0; k < Four_D_array[0][0][0].length; k++)
           Four_D_array[i][j][k][a] = i+j+k+a;
for(int  a = 0; a < Four_D_array.length; a++)                    {
  for(int i = 0; i < Four_D_array.length[0]; i++)   {
     for(int j = 0; j < Four_D_array[0][0].length; j++)   {
        for(int k = 0; k < Four_D_array[0][0][0].length; k++)
           System.out.print(Four_D_array[i][j][k][a]+ " ");
      System.out.println();                                  }
   System.out.println();                              }
 System.out.println();                                        }
```

```
Four dimentional array
0 1 2 3 4
1 2 3 4 5
2 3 4 5 6
3 4 5 6 7

1 2 3 4 5
2 3 4 5 6
3 4 5 6 7
4 5 6 7 8

2 3 4 5 6
3 4 5 6 7
4 5 6 7 8
5 6 7 8 9


1 2 3 4 5
2 3 4 5 6
3 4 5 6 7
4 5 6 7 8

2 3 4 5 6
3 4 5 6 7
4 5 6 7 8
5 6 7 8 9

3 4 5 6 7
4 5 6 7 8
5 6 7 8 9
6 7 8 9 10
```

# *Arrays* de *Strings*

A declaração de *array* de *Strings* cria uma sequência de referências nulas para *String* que depois serão preenchidas.

```
import java.util.*;

public class ArrayStrings {

  static Scanner read = new Scanner(System.in);
public static void main (String args[])  {
  String nomes[] = new String[5];
  for(int i=0; i < nomes.length; i++)
      nomes[i] = read.nextLine();
  print(nomes);
}
public static void print(String s[])
{             for(int i = 0; i < s.length; i++)
                System.out.printf("s[%d] = %s\n",i,s[i]);                    }

}
```

```
Paulo
Ana
Nuno
Rui
Ricardo
s[0] = Paulo
s[1] = Ana
s[2] = Nuno
s[3] = Rui
s[4] = Ricardo
Press any key to continue . . .
```

```
import java.util.*;

public class ArrayStrings {
            static Scanner read = new Scanner(System.in);
public static void main (String args[])  {
  String    nomes[] = new String[5];
  int       inteiros[] = new int[5];
  for(int i=0; i < nomes.length; i++)       {
            System.out.print("Next int   ?   ");
            inteiros[i] = read.nextInt();
            System.out.print("Next line   ?   ");
            nomes[i] = read.nextLine();                    }
  print(nomes);
}
public static void print(String s[])
{           for(int i = 0; i < s.length; i++)
               System.out.printf("s[%d] = %s\n",i,s[i]);                    }

}
```

*Funcionamento inesperável !!!*

```
Next int    ?   5
Next line    ?   Next int    ?   6
Next line    ?   Next int    ?   7
Next line    ?   Next int    ?   8
Next line    ?   Next int    ?   5
Next line    ?   s[0] =
s[1] =
s[2] =
s[3] =
s[4] =
Press any key to continue . . .
```

# Solução

```
import java.util.*;

public class ArrayStrings {
            static Scanner read = new Scanner(System.in);
public static void main (String args[])  {
  String    nomes[] = new String[5];
  int       inteiros[] = new int[5];
  for(int i=0; i < nomes.length; i++)       {
            System.out.print("Next int   ?   ");
            inteiros[i] = read.nextInt();
            System.out.print("Next line   ?   ");
            do nomes[i] = read.nextLine(); while(nomes[i].length() == 0);
                                        }
  print(nomes);
}
public static void print(String s[])
{           for(int i = 0; i < s.length; i++)
               System.out.printf("s[%d] = %s\n",i,s[i]);                    }

}
```

*Ok*

```
Next int   ?   1
Next line   ?   Aveiro
Next int   ?   2
Next line   ?   Braga
Next int   ?   3
Next line   ?   Coimbra
Next int   ?   4
Next line   ?   Porto
Next int   ?   5
Next line   ?   Lisboa
s[0] = Aveiro
s[1] = Braga
s[2] = Coimbra
s[3] = Porto
s[4] = Lisboa
Press any key to continue . . .
```

# *Arrays* de *Strings*

Ordenação de *Strings*.

```
import java.util.*;

public class ArrayStringsSort {

  static Scanner read = new Scanner(System.in);
public static void main (String args[])  {
   String cidades[] = new String[7];
   for(int i=0; i < cidades.length; i++)
      cidades[i] = read.nextLine();
   Arrays.sort(cidades);
   print(cidades);
}
public static void print(String s[])
{              for(int i = 0; i < s.length; i++)
                  System.out.printf("s[%d] = %s\n",i,s[i]);                    }

}
```

```
Lisbon
Porto
Aveiro
Coimbra
Braga
Faro
Covilha
s[0] = Aveiro
s[1] = Braga
s[2] = Coimbra
s[3] = Covilha
s[4] = Faro
s[5] = Lisbon
s[6] = Porto
Press any key to continue . . .
```

Pode ordenar Strings utilizando a função sort da classe Array

Cuidado! Na avaliação, dependente das condições de tarefas, esta função pode ser permitida ou não

Deve perguntar

## *Arrays de objetos de uma classe (arrays de registos)*

```
import java.util.*;        import java.io.*;
public class ArrayDeObjetos {
  static Scanner read = new Scanner(System.in);
  static final int N = 5;  // número máximo de alunos numa turma
public static void main (String args[])   throws IOException     {
   aluno alunos[] = new aluno[N];
   File students = new File("File_with_students.txt");
   PrintWriter ws = new PrintWriter(students);
   for(int i = 0; i < alunos.length; i++)
   {                alunos[i] = new aluno();
                    System.out.print("Nome  ?  ");
                    alunos[i].nome = read.nextLine();
                    ws.print(alunos[i].nome+" ");
                    System.out.print("N mec   ?  ");
                    alunos[i].n_mec = read.nextInt();
                    ws.print(alunos[i].n_mec+" ");
                    System.out.print("Nota media   ?  ");
                    alunos[i].nota_media = read.nextDouble();
                    ws.print(alunos[i].nota_media+" ");
                    System.out.print("Idade   ?  ");
                    alunos[i].idade = read.nextInt();
                    ws.print(alunos[i].idade+" ");
                    ws.println();
                    while (read.nextLine().length() != 0);
    }
    ws.close();
}   }
```

```
class aluno {
            String   nome;
            int      n_mec;
            double nota_media;
            int      idade;
}
```

Vai ser criado um ficheiro com informação sobre alunos

```
Nome   ?  Nuno
N mec    ?  6547
Nota media    ?  15.3
Idade    ?  21
Nome   ?  Claudia
N mec    ?  3454
Nota media    ?  12.9
Idade    ?  20
Nome   ?  Pedro
N mec    ?  5566
Nota media    ?  14.5
Idade    ?  22
Nome   ?  Luis
N mec    ?  4910
Nota media    ?  9.6
Idade    ?  25
Nome   ?  Ana
N mec    ?  3354
Nota media    ?  13.0
Idade    ?  21
Press any key to continue . . .
```

## *Arrays de objetos de uma classe (arrays de registos)*

```
import java.util.*;        import java.io.*;
public class ArrayDeObjetos {
  static Scanner read = new Scanner(System.in);
  static final int N = 5;  // número máximo de alunos numa turma
public static void main (String args[])   throws IOException     {
  aluno alunos[] = new aluno[N];
  File students = new File("File_with_students.txt");
  PrintWriter ws = new PrintWriter(students);
  for(int i = 0; i < alunos.length; i++)
  {                alunos[i] = new aluno();
                   System.out.print("Nome  ?  ");
                   alunos[i].nome = read.nextLine();
                   ws.print(alunos[i].nome+" ");
                   System.out.print("N mec   ?  ");
                   alunos[i].n_mec = read.nextInt();
                   ws.print(alunos[i].n_mec+" ");
                   System.out.print("Nota media   ?  ");
                   alunos[i].nota_media = read.nextDouble();
                   ws.print(alunos[i].nota_media+" ");
                   System.out.print("Idade   ?  ");
                   alunos[i].idade = read.nextInt();
                   ws.print(alunos[i].idade+" ");
                   ws.println();
                   while (read.nextLine().length() != 0);
   }
   ws.close();
}   }
```

Declarar um *array* de registos:

```
class aluno {
        String   nome;
        int      n_mec;
        double nota_media;
        int      idade;
}
```

Para escrever os elementos do *array* num ficheiro

Escrever vários campos do objeto em ficheiro File_with_students.txt

Ficheiro *File_with_students.txt*

```
1  Nuno 6547 15.3 21
2  Claudia 3454 12.9 20
3  Pedro 5566 14.5 22
4  Luis 4910 9.6 25
5  Ana 3354 13.0 21
6
```

# *Arrays de objetos de uma classe (**arrays de registos**)*

Vai ser criado um ficheiro com informação sobre alunos

```
Nome   ?  Nuno
N mec    ?  6547
Nota media    ?  15.3
Idade    ?  21
Nome   ?  Claudia
N mec    ?  3454
Nota media    ?  12.9
Idade    ?  20
Nome   ?  Pedro
N mec    ?  5566
Nota media    ?  14.5
Idade    ?  22
Nome   ?  Luis
N mec    ?  4910
Nota media    ?  9.6
Idade    ?  25
Nome   ?  Ana
N mec    ?  3354
Nota media    ?  13.0
Idade    ?  21
Press any key to continue . . .
```

```
File students = new File("File_with_students.txt");
PrintWriter ws = new PrintWriter(students);
```

Ficheiro *File_with_students.txt*

```
1  Nuno 6547 15.3 21
2  Claudia 3454 12.9 20
3  Pedro 5566 14.5 22
4  Luis 4910 9.6 25
5  Ana 3354 13.0 21
6
```

# *Arrays de objetos de uma classe (**arrays de registos**)*

```
import java.util.*;  import java.io.*;
public class SortDifferentFields {
  static Scanner read = new Scanner(System.in);
  static final int N = 5;  // número máximo de alunos numa turma
public static void main (String args[])   throws IOException    {
   String names_of_students[] = new String[N];
   File students = new File("File_with_students.txt");
   Scanner stud = new Scanner(students);
   File sort_names = new File("sorted_by_names.txt");
   PrintWriter wsn = new PrintWriter(sort_names);
   for(int i = 0; i < N; i++)
              names_of_students[i] = stud.nextLine();
   Arrays.sort(names_of_students);
   for(int i = 0; i < N; i++)
       wsn.printf("%s\n",names_of_students[i]);
    stud.close();
    wsn.close();
}
}
```

*Ordenar linhas por nomes de alunos*

Ficheiro *File_with_students.txt*

```
1  Nuno 6547 15.3 21
2  Claudia 3454 12.9 20
3  Pedro 5566 14.5 22
4  Luis 4910 9.6 25
5  Ana 3354 13.0 21
6
```

Ficheiro *sorted_by_names.txt*

```
1  Ana 3354 13.0 21
2  Claudia 3454 12.9 20
3  Luis 4910 9.6 25
4  Nuno 6547 15.3 21
5  Pedro 5566 14.5 22
```

Para reduzir o tamanho do código várias verificações não foram feitas (mas devem ser feitas)

## *Arrays* de objetos de uma classe (*arrays* de registos)  *Ordenar linhas por nomes de alunos*

```
import java.util.*;  import java.io.*;
public class SortDifferentFields {
  static Scanner read = new Scanner(System.in);
  static final int N = 5;  // número máximo de alunos numa turma
public static void main (String args[])   throws IOException     {
  String names_of_students[] = new String[N];
  File students = new File("File_with_students.txt");
  Scanner stud = new Scanner(students);
  File sort_names = new File("sorted_by_names.txt");
  PrintWriter wsn = new PrintWriter(sort_names);
  for(int i = 0; i < N; i++)
              names_of_students[i] = stud.nextLine();
  Arrays.sort(names_of_students);
  for(int i = 0; i < N; i++)
      wsn.printf("%s\n",names_of_students[i]);
   stud.close();
   wsn.close();
}
}
```

Declarar um array de Strings

Para ler dados do ficheiro File_with_students.txt

Para escrever linhas com nomes ordenados num ficheiro sorted_by_names.txt

Para ordenação

Ficheiro *sorted_by_names.txt*

```
1  Ana 3354 13.0 21
2  Claudia 3454 12.9 20
3  Luis 4910 9.6 25
4  Nuno 6547 15.3 21
5  Pedro 5566 14.5 22
```

Para reduzir o tamanho do código várias verificações não foram feitas (mas devem ser feitas)

## *Arrays de objetos de uma classe (**arrays de registos**)*

*Ordenar linhas por nomes de alunos*

```
import java.util.*;  import java.io.*;
public class SortDifferentFields {
  static Scanner read = new Scanner(System.in);
  static final int N = 5;  // número máximo de alunos numa turma
public static void main (String args[])   throws IOException     {
   String names_of_students[] = new String[N];
   File students = new File("File_with_students.txt");
   if ( !verify(students,"File_with_students.txt") ) { System.out.println("E R R O R "); System.exit(1); }
   Scanner stud = new Scanner(students);
   File sort_names = new File("sorted_by_names.txt");
   PrintWriter wsn = new PrintWriter(sort_names);
   for(int i = 0; i < N; i++)   names_of_students[i] = stud.nextLine();
   Arrays.sort(names_of_students);
   for(int i = 0; i < N; i++)   wsn.printf("%s\n",names_of_students[i]);
    stud.close();               wsn.close();
}

static boolean verify(File my_file, String nameIn)
{  if (!my_file.exists())              {
                                 System.out.println("ERROR: input file " + nameIn + " does not exist!");        return false;      }
  if (my_file.isDirectory())       {
                                 System.out.println("ERROR: input file " + nameIn + " is a directory!");        return false;      }
  if (!my_file.canRead())        {
                                 System.out.println("ERROR: cannot read from input file " + nameIn+ "!");     return false;      }
   return true;                                                    }
}
```

Exemplo com verificação

# *Arrays* de objetos de uma classe (*arrays* de registos)

```java
import java.util.*;        import java.io.*;
public class SortDifferentFields1 {
  static Scanner read = new Scanner(System.in);
  static final int N = 5;  // número máximo de alunos numa turma
public static void main (String args[])   throws IOException     {
   String names_of_students[] = new String[N];
   int n_mecs[] = new int[N];   double notas[] = new double[N];
   int idades[] = new int[N];
   File students = new File("File_with_students.txt");
   Scanner stud = new Scanner(students);
   File sort_names = new File("sorted_by_names.txt");
   PrintWriter wsn = new PrintWriter(sort_names);
   for(int i = 0; i < N; i++)                {
       names_of_students[i] = stud.next();
       while (!stud.hasNextInt()) names_of_students[i] += stud.next();
       n_mecs[i] = stud.nextInt();   notas[i] = stud.nextDouble();
       idades[i] = stud.nextInt();        }
    wsn.printf("\nSort by names: \n");
    Arrays.sort(names_of_students);
    for(int i = 0; i < N; i++)  wsn.printf("%s\n",names_of_students[i]);
    wsn.printf("\nSort by n_mec: \n");
    Arrays.sort(n_mecs);
    for(int i = 0; i < N; i++)  wsn.printf("%s\n",n_mecs[i]);
    wsn.printf("\nSort by grades: \n");
    Arrays.sort(notas);
    for(int i = 0; i < N; i++)  wsn.printf("%s\n",notas[i]);
    wsn.printf("\nSort by age: \n");
    Arrays.sort(idades);
    for(int i = 0; i < N; i++)  wsn.printf("%s\n",idades[i]);
    stud.close();   wsn.close();
} }
```

e-learning

Preparar *arrays* para ordenar

Ordenar *arrays* e escrever os resultados no ficheiro sorted_by_names.txt

Fechar ficheiros

Para reduzir o tamanho do código várias verificações não foram feitas (mas devem ser feitas)

Ficheiro *File_with_students.txt*

```
Nuno 6547 15.3 21
Claudia 3454 12.9 20
Pedro 5566 14.5 22
Luis 4910 9.6 25
Ana 3354 13.0 21
```

```
Sort by names:
Ana
Claudia
Luis
Nuno
Pedro

Sort by n_mec:
3354
3454
4910
5566
6547

Sort by grades:
9.6
12.9
13.0
14.5
15.3

Sort by age:
20
21
21
22
25
```

# Programa final que já tem complexidade para tarefas práticas

Valeri Skliarov
2016/2017

***Tarefa***: Usar o ficheiro File_with_students.txt" e preparar um ficheiro novo File_with_students_sorted_by_different_fields.txt do tipo seguinte:

```
Sort by names:
  Alexandre Carvalho   3499   16.3   30
        Ana Patricia   4511   13.7   22
       Nuno Ferreira   3671   15.4   23
         Pedro Silva   7619   10.9   19
        Ricardo reis   9921   17.1   27

Sort by n_mec:
  Alexandre Carvalho   3499   16.3   30
       Nuno Ferreira   3671   15.4   23
        Ana Patricia   4511   13.7   22
         Pedro Silva   7619   10.9   19
        Ricardo reis   9921   17.1   27

Sort by grades:
         Pedro Silva   7619   10.9   19
        Ana Patricia   4511   13.7   22
       Nuno Ferreira   3671   15.4   23
  Alexandre Carvalho   3499   16.3   30
        Ricardo reis   9921   17.1   27

Sort by age:
         Pedro Silva   7619   10.9   19
        Ana Patricia   4511   13.7   22
       Nuno Ferreira   3671   15.4   23
        Ricardo reis   9921   17.1   27
  Alexandre Carvalho   3499   16.3   30
```

```java
import java.util.*;        import java.io.*;
public class ArrayDeObjetosAligned {
  static Scanner read = new Scanner(System.in);
  static final int N = 7;  // número máximo de alunos numa turma
public static void main (String args[])   throws IOException     {
   aluno alunos[] = new aluno[N];
   File students = new File("Aligned_file_with_students.txt");
   PrintWriter ws = new PrintWriter(students);
   for(int i = 0; i < alunos.length; i++)
   {                alunos[i] = new aluno();
                    System.out.print("Nome  ?  ");
                    alunos[i].nome = read.nextLine();
                    ws.printf("%20s",alunos[i].nome);
                    System.out.print("N mec   ?  ");
                    alunos[i].n_mec = read.nextInt();
                    ws.printf("%7d",alunos[i].n_mec);
                    System.out.print("Nota media   ?  ");
                    alunos[i].nota_media = read.nextDouble();
                    ws.printf("%7.1f",alunos[i].nota_media);
                    System.out.print("Idade   ?  ");
                    alunos[i].idade = read.nextInt();
                    ws.printf("%4d",alunos[i].idade);
                    ws.println();
                    while (read.nextLine().length() != 0);
   }
   ws.close();
} }
```

Alinhar vários campos

```java
class aluno {
                String nome;
                int n_mec;
                double nota_media;
                int idade;
}
```

```
Nome   ?  Nuno Ferreira
N mec    ?  3671
Nota media    ?  15.4
Idade    ?  23
Nome   ?  Ana Patricia
N mec    ?  4511
Nota media    ?  13.7
Idade    ?  22
Nome   ?  Alexandre Carvalho
N mec    ?  3499
Nota media    ?  16.3
Idade    ?  30
Nome   ?  Pedro Silva
N mec    ?  7619
Nota media    ?  10.9
Idade    ?  19
Nome   ?  Ricardo reis
N mec    ?  9921
Nota media    ?  17.1
Idade    ?  27
Nome   ?  Claudia Silva
N mec    ?  5676
Nota media    ?  14
Idade    ?  20
Nome   ?  Carla Pereira
N mec    ?  4691
Nota media    ?  16.2
Idade    ?  28
Press any key to continue . . .
```

```
Nome   ?  Nuno Ferreira
N mec    ?  3671
Nota media    ?  15.4
Idade    ?  23
Nome   ?  Ana Patricia
N mec    ?  4511
Nota media    ?  13.7
Idade    ?  22
Nome   ?  Alexandre Carvalho
N mec    ?  3499
Nota media    ?  16.3
Idade    ?  30
Nome   ?  Pedro Silva
N mec    ?  7619
Nota media    ?  10.9
Idade    ?  19
Nome   ?  Ricardo reis
N mec    ?  9921
Nota media    ?  17.1
Idade    ?  27
Nome   ?  Claudia Silva
N mec    ?  5676
Nota media    ?  14
Idade    ?  20
Nome   ?  Carla Pereira
N mec    ?  4691
Nota media    ?  16.2
Idade    ?  28
Press any key to continue . . .
```

Ficheiro

Aligned_file_with_students.txt

que vai ser criado

```
      Nuno Ferreira   3671   15.4   23
       Ana Patricia   4511   13.7   22
Alexandre Carvalho    3499   16.3   30
       Pedro Silva    7619   10.9   19
       Ricardo reis   9921   17.1   27
      Claudia Silva   5676   14.0   20
      Carla Pereira   4691   16.2   28
```

***Tarefa***: Usar o ficheiro Aligned_file_with_students.txt " e preparar um ficheiro novo File_with_students_sorted_by_different_fields.txt do tipo seguinte:

Ficheiro
Aligned_file_with_students.txt
que vai ser criado

| | | | |
|---|---|---|---|
| Nuno Ferreira | 3671 | 15.4 | 23 |
| Ana Patricia | 4511 | 13.7 | 22 |
| Alexandre Carvalho | 3499 | 16.3 | 30 |
| Pedro Silva | 7619 | 10.9 | 19 |
| Ricardo reis | 9921 | 17.1 | 27 |
| Claudia Silva | 5676 | 14.0 | 20 |
| Carla Pereira | 4691 | 16.2 | 28 |

```
Sort by names:
  Alexandre Carvalho   3499   16.3   30
        Ana Patricia   4511   13.7   22
       Carla Pereira   4691   16.2   28
       Claudia Silva   5676   14.0   20
       Nuno Ferreira   3671   15.4   23
         Pedro Silva   7619   10.9   19
        Ricardo reis   9921   17.1   27

Sort by n_mec:
  Alexandre Carvalho   3499   16.3   30
       Nuno Ferreira   3671   15.4   23
        Ana Patricia   4511   13.7   22
       Carla Pereira   4691   16.2   28
       Claudia Silva   5676   14.0   20
         Pedro Silva   7619   10.9   19
        Ricardo reis   9921   17.1   27

Sort by grades:
         Pedro Silva   7619   10.9   19
        Ana Patricia   4511   13.7   22
       Claudia Silva   5676   14.0   20
       Nuno Ferreira   3671   15.4   23
       Carla Pereira   4691   16.2   28
  Alexandre Carvalho   3499   16.3   30
        Ricardo reis   9921   17.1   27

Sort by age:
         Pedro Silva   7619   10.9   19
       Claudia Silva   5676   14.0   20
        Ana Patricia   4511   13.7   22
       Nuno Ferreira   3671   15.4   23
        Ricardo reis   9921   17.1   27
       Carla Pereira   4691   16.2   28
  Alexandre Carvalho   3499   16.3   30
```

**Não usar funções de ordenação das bibliotecas de java**

# Ideias principais:

1. Extrair campos para ordenação que são *nomes*, *números mecanográficos*, *notas* e *idades*.
2. Criar *arrays* de índices com valores 0,1,…,N-1 para cada campo que deve ser ordenado.
3. Ordenar cada campo e quando trocar os elementos trocar também os índices relevantes.
4. Gravar os dados no ficheiro utilizando valores dos *arrays* de índices como índices das linhas.

# Exemplo:

```
Nuno Ferreira     3671    15.4   23
Ana Patricia      4511    13.7   22
Alexandre Carvalho 3499   16.3   30
Pedro Silva       7619    10.9   19
Ricardo reis      9921    17.1   27
Claudia Silva     5676    14.0   20
Carla Pereira     4691    16.2   28
```

```
Sort by names:
  Alexandre Carvalho   3499   16.3   30
        Ana Patricia   4511   13.7   22
       Carla Pereira   4691   16.2   28
       Claudia Silva   5676   14.0   20
       Nuno Ferreira   3671   15.4   23
         Pedro Silva   7619   10.9   19
        Ricardo reis   9921   17.1   27

Sort by n_mec:
  Alexandre Carvalho   3499   16.3   30
       Nuno Ferreira   3671   15.4   23
        Ana Patricia   4511   13.7   22
       Carla Pereira   4691   16.2   28
       Claudia Silva   5676   14.0   20
         Pedro Silva   7619   10.9   19
        Ricardo reis   9921   17.1   27

Sort by grades:
         Pedro Silva   7619   10.9   19
        Ana Patricia   4511   13.7   22
       Claudia Silva   5676   14.0   20
       Nuno Ferreira   3671   15.4   23
       Carla Pereira   4691   16.2   28
  Alexandre Carvalho   3499   16.3   30
        Ricardo reis   9921   17.1   27

Sort by age:
         Pedro Silva   7619   10.9   19
       Claudia Silva   5676   14.0   20
        Ana Patricia   4511   13.7   22
       Nuno Ferreira   3671   15.4   23
        Ricardo reis   9921   17.1   27
       Carla Pereira   4691   16.2   28
  Alexandre Carvalho   3499   16.3   30
```

# Exemplo:

Valores do *array* de índices iniciais

```
for(int j = 0; j < N; j++) {
            names_of_students_in[j] = j;
            n_mecs_in[j] = j;
            notas_in[j] = j;
            idades_in[j] = j;      }
```

Função considerada na aula com pequenas alterações

```
wsn.printf("\nSort by age: \n");
sortCrescSeq(idades, idades_in);
for(int i = 0; i < N; i++)
      wsn.printf("%s\n",line[ idades_in[i] ]);
```

Gravar ficheiro utilizando índices como valores do *array* de índicas

```
Sort by age:
        Pedro Silva    7619    10.9   19
      Claudia Silva    5676    14.0   20
       Ana Patricia    4511    13.7   22
      Nuno Ferreira    3671    15.4   23
       Ricardo reis    9921    17.1   27
      Carla Pereira    4691    16.2   28
  Alexandre Carvalho   3499    16.3   30
```

Declaração de line:

```
String line[] = new String[N];
```

```
sortCrescSeq(idades, idades_in);
```

**Ordenar dados**

```
public static void sortCrescSeq(int[] num_array, int[] index) {
  for(int i = 0;i < num_array.length-1;i++)
    for(int j = i + 1;j < num_array.length;j++)
        comparar_trocar(num_array,index,j,i);          }
```

**Trocar valores**

```
public static void comparar_trocar(int a[],int index[], int indice1, int indice2)   {
    if (a[indice1] < a[indice2])
    {   int tmp = a[indice1];
        a[indice1] = a[indice2];
        a[indice2] = tmp;
        exchange_index(index,indice1,indice2);   }
                }
```

**Trocar valores no *array* de índices**

```
public static void exchange_index(int a[], int indice1, int indice2) {
        int tmp = a[indice1];
        a[indice1] = a[indice2];
        a[indice2] = tmp;                                    }
```

Trocar valores no *array* de índices

# Funções

```
public static void sortCrescSeq(String[] num_array, int[] index) {
 for(int i = 0;i < num_array.length-1;i++)
   for(int j = i + 1;j < num_array.length;j++)
       comparar_trocar(num_array,index,j,i);          }
public static void sortCrescSeq(int[] num_array, int[] index) {
 for(int i = 0;i < num_array.length-1;i++)
   for(int j = i + 1;j < num_array.length;j++)
       comparar_trocar(num_array,index,j,i);          }
public static void sortCrescSeq(double[] num_array, int[] index) {
 for(int i = 0;i < num_array.length-1;i++)
   for(int j = i + 1;j < num_array.length;j++)
       comparar_trocar(num_array,index,j,i);          }
```

Sobrecarga do nome da função sortCrescSeq para processar *Strings*, inteiros e reais

```
public static void comparar_trocar(String a[],int index[], int indice1, int indice2)   {
   if (a[indice1].compareTo(a[indice2]) < 0)
   {  String tmp = a[indice1];   a[indice1] = a[indice2];  a[indice2] = tmp; // troca de valores
      exchange_index(index,indice1,indice2);   }                              // troca de índices
                }
public static void comparar_trocar(int a[],int index[], int indice1, int indice2)   {
   if (a[indice1] < a[indice2])
   {  int tmp = a[indice1];  a[indice1] = a[indice2]; a[indice2] = tmp;       // troca de valores
      exchange_index(index,indice1,indice2);   }                              // troca de índices
                }
public static void comparar_trocar(double a[],int index[], int indice1, int indice2)   {
   if (a[indice1] < a[indice2])
   {  double tmp = a[indice1];  a[indice1] = a[indice2];  a[indice2] = tmp; // troca de valores
      exchange_index(index,indice1,indice2);   }                              // troca de índices
                                                }
public static void exchange_index(int a[], int indice1, int indice2) {
      int tmp = a[indice1];       a[indice1] = a[indice2];          a[indice2] = tmp;
}
```

Sobrecarga do nome da função comparar_trocar para processar *Strings*, inteiros e reais

# Função main

```
public static void main (String args[])   throws IOException    {
  String line[] = new String[N];   String names_of_students[] = new String[N];
  int n_mecs[] = new int[N];   double notas[] = new double[N];
  int idades[] = new int[N];    int names_of_students_in[] = new int[N];
  int n_mecs_in[] = new int[N];   int notas_in[] = new int[N];   int idades_in[] = new int[N];
  for(int j = 0; j < N; j++) {
                  names_of_students_in[j] = j;   n_mecs_in[j] = j;
                  notas_in[j] = j;                idades_in[j] = j;      }
  File students = new File("Aligned_file_with_students.txt");
  Scanner stud = new Scanner(students);
  File sorted = new File("File_with_students_sorted_by_different_fields.txt");
  PrintWriter wsn = new PrintWriter(sorted);
  for(int i = 0; i < N; i++)   line[i] = stud.nextLine();
  stud.close();
  stud = new Scanner(students);
  for(int i = 0; i < N; i++)  {
     names_of_students[i] = stud.next();
     while (!stud.hasNextInt()) names_of_students[i] += stud.next();
     n_mecs[i] = stud.nextInt();   notas[i] = stud.nextDouble();   idades[i] = stud.nextInt();      }
  wsn.printf("\nSort by names: \n");
  sortCrescSeq(names_of_students, names_of_students_in);
  for(int i = 0; i < N; i++)     wsn.printf("%s\n",line[ names_of_students_in[i] ]);
  wsn.printf("\nSort by n_mec: \n");
  sortCrescSeq(n_mecs, n_mecs_in);
  for(int i = 0; i < N; i++)     wsn.printf("%s\n",line[ n_mecs_in[i] ]);
  wsn.printf("\nSort by grades: \n");
  sortCrescSeq(notas, notas_in);
  for(int i = 0; i < N; i++)     wsn.printf("%s\n",line[ notas_in[i] ]);
  wsn.printf("\nSort by age: \n");
  sortCrescSeq(idades, idades_in);
  for(int i = 0; i < N; i++)     wsn.printf("%s\n",line[ idades_in[i] ]);
  stud.close();   wsn.close();   }
```

Arrays de índices

Inicialização do *arrays* de índices

Para ficheiros

Leitura das linhas do ficheiro

Fechar stud e abrir stud novamente

Leitura dos campos do ficheiro

Linhas com vários campos ordenados

e-learning

Para reduzir o tamanho do código várias verificações não foram feitas (mas devem ser feitas)

# Programa completo

```java
import java.util.*;          import java.io.*;
public class SortStudentsByDifferentFields {
  static Scanner read = new Scanner(System.in);
  static final int N = 7;  // numero de alunos maximo numa turma
public static void main (String args[])   throws IOException     {
   String line[] = new String[N];
   String names_of_students[] = new String[N];
   int n_mecs[] = new int[N];
   double notas[] = new double[N];
   int idades[] = new int[N];
   int names_of_students_in[] = new int[N];
   int n_mecs_in[] = new int[N];
   int notas_in[] = new int[N];
   int idades_in[] = new int[N];
   for(int j = 0; j < N; j++) {
       names_of_students_in[j] = j;
       n_mecs_in[j] = j;
       notas_in[j] = j;
       idades_in[j] = j;        }
   File students = new File("Aligned_file_with_students.txt");
   Scanner stud = new Scanner(students);
   File sorted = new File("File_with_students_sorted_by_different_fields.txt");
   PrintWriter wsn = new PrintWriter(sorted);
   for(int i = 0; i < N; i++)    line[i] = stud.nextLine();
   stud.close();
   stud = new Scanner(students);
   for(int i = 0; i < N; i++)  {
        names_of_students[i] = stud.next();
        while (!stud.hasNextInt()) names_of_students[i] += stud.next();
        n_mecs[i] = stud.nextInt();
        notas[i] = stud.nextDouble();
        idades[i] = stud.nextInt();         }
    wsn.printf("\nSort by names: \n");
    sortCrescSeq(names_of_students, names_of_students_in);
    for(int i = 0; i < N; i++)
        wsn.printf("%s\n",line[ names_of_students_in[i] ]);
    wsn.printf("\nSort by n_mec: \n");
    sortCrescSeq(n_mecs, n_mecs_in);
    for(int i = 0; i < N; i++)
        wsn.printf("%s\n",line[ n_mecs_in[i] ]);
    wsn.printf("\nSort by grades: \n");
    sortCrescSeq(notas, notas_in);
    for(int i = 0; i < N; i++)
        wsn.printf("%s\n",line[ notas_in[i] ]);
    wsn.printf("\nSort by age: \n");
    sortCrescSeq(idades, idades_in);
    for(int i = 0; i < N; i++)
        wsn.printf("%s\n",line[ idades_in[i] ]);
    stud.close();
    wsn.close();
}
public static void sortCrescSeq(String[] num_array, int[] index) {
public static void sortCrescSeq(int[] num_array, int[] index) {
public static void sortCrescSeq(double[] num_array, int[] index) {
public static void comparar_trocar(String a[],int index[], int indice1, int indice2)    {
public static void comparar_trocar(int a[],int index[], int indice1, int indice2)    {
public static void comparar_trocar(double a[],int index[], int indice1, int indice2)    {
public static void exchange_index(int a[], int indice1, int indice2) { int tmp = a[indice1];
}
```

Para reduzir o tamanho do código várias verificações não foram feitas (mas devem ser feitas)

## Exercício 12.4

O programa `ProcessImage.java` permite ler uma imagem, num ficheiro em formato Plain PGM, e produz uma nova imagem modificada, que grava no ficheiro `out.pgm`. Compile e experimente esse programa sobre uma das imagens fornecidas. Pode visualizar as imagens com o programa `eog` (Eye-of-gnome) ou outro.

Crie uma função que rode a imagem de 90º no sentido dos ponteiros do relógio. Modifique o programa para fazer essa operação.

**Note:** O formato Plain PGM é pouco eficiente e por isso é raramente usado. Num sistema Linux pode usar o comando `convert` (do pacote ImageMagik) para converter entre formatos de imagem.

**PGM** é acrónimo de "**P**ortable **G**ray **M**ap"

Pode também abrir ficheiros deste formato em Windows IrfanView

Os slides seguintes permitem perceber como criar e verificar ficheiros no formato PGM

## Exemplo 1:

| Name | Date modified | Type | Size |
|---|---|---|---|
| trivial.pgm | 12/8/2014 3:16 PM | IrfanView PGM File | 18 KB |
| TrivialImage.class | 12/8/2014 3:16 PM | CLASS File | 2 KB |
| TrivialImage.java | 12/8/2014 3:16 PM | JAVA File | 1 KB |

```
import java.util.*;     // exemplo 1
import java.io.*;
public class TrivialImage {
  static Scanner read = new Scanner(System.in);
public static void main (String args[])  throws IOException     {
int largura, altura;
File fout = new File("trivial.pgm");
PrintWriter pw = new PrintWriter(fout);
pw.println("P2");                          // tipo
pw.println("30 150");                      // largura, altura
pw.println("255 ");                        // intensidade máxima

  for(int i = 0; i<150; i++)  {
     for(int j = 0; j<30; j++)
        if ( j<10 | | j>25 ) pw.print ("255 "); // duas linhas verticais do lado esquerdo e direito
        else pw.print ("100 ");
     pw.printf ("\n");          }
pw.close();
}      }
```

e-learning

```
for(int i = 0; i<150; i++)  {
   for(int j = 0; j<30; j++)
      if ( j<10 || j>25 ) pw.print ("255 ");
      else pw.print ("100 ");
   pw.printf ("\n");          }
```

Notepad++

H:\MostFrequentlyNeeded__2014\Education2011\2014_2015\FirstSemester\Programacao\AulasTeoricas\12_9_12\Examples\image\trivial.pgm - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

copy.c | copy.h | zed_64.bin | Nov2014zed_64_b256.bin | My7.pgm | zed_64_b256.bin | File_with_students.txt | sorted_by_names.txt | File_with_students_so

```
P2
30 150
255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255
```

```
pw.println("P2");                    // tipo
pw.println("30 150");                // largura, altura
pw.println("255 ");                  // intensidade máxima
```

## Exemplo 2:

```
import java.util.*;                    // exemplo 2
import java.io.*;
public class TrivialImageInten {
  static Scanner read = new Scanner(System.in);
public static void main (String args[])  throws IOException     {
int largura, altura;
File fout = new File("trivialInten.pgm");
PrintWriter pw = new PrintWriter(fout);
pw.println("P2");                      // tipo
pw.println("30 255");                  // largura, altura
pw.println("255 ");                    // intendidade máxima

  for(int i = 0; i<255; i++)  {
     for(int j = 0; j<30; j++)
        pw.printf ("%s ",i);      // Alteração de intensidade
     pw.printf ("\n");           }
pw.close();
}      }
```

e-learning

```
P2
30 255
255
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10
11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11
12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12
13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13
14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14
15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15
16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16
17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17
18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18
19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21 21
22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22
23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23 23
24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24
25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25
26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26
27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27 27
```

Notepad++

```
for(int i = 0; i<255; i++)  {
    for(int j = 0; j<30; j++)
        pw.printf ("%s ",i);      // Alteração de intensidade
    pw.printf ("\n");           }
```

## Exemplo 3:

```
import java.util.*;                                  // exemplo 3
import java.io.*;
public class TrivialImage1 {
  static Scanner read = new Scanner(System.in);
public static void main (String args[])  throws IOException
int largura, altura;
File fout = new File("trivial1.pgm");
PrintWriter pw = new PrintWriter(fout);
pw.println("P2");                    // tipo
pw.println("500 300");               // largura, altura
pw.println("255 ");                  // intensidade máxima

  for(int i = 0; i<=300; i++)
     for(int j = 0; j<500; j++)
        if (i<50 || i>250) pw.print ("255 ");
        else if (j<50 || j>450)  pw.print ("255 ");
        else    pw.print ("100 ");
pw.close();
}      }
```

e-learning

# Exemplo 4:

Esta imagem permite desenhar a figura seguinte

```
P2
30 43
255
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100
100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 255 255 255 100 100 100 100 100 255 255 255 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 255 255 255 100 100 100 100 100 255 255 255 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 255 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 255 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 255 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 255 255 255 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 255 100 100 100 100 100 255 255 255 255 255 255 255 255 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 100 255 100 100 100 100 100 255 255 255 255 255 255 255 100 100 100 100 255 100 100 100 100 100 100
100 100 100 100 100 100 100 255 100 100 100 100 100 100 255 255 100 100 100 100 100 255 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 255 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
```

## Exemplo 4:

```
import java.util.*;                                  // exemplo 4
import java.io.*;
public class AlterarFaceImage {
public static void main (String args[])  throws IOException     {
int largura, altura;    String tmp = new String();
File orid = new File("face_image.pgm");
Scanner rd = new Scanner(orid);
File fout = new File("changed_face_image.pgm");
PrintWriter pw = new PrintWriter(fout);
pw.println("P2");                    // tipo
pw.println("30 43");                 // largura, altura
pw.println("255 ");                  // intendidade máxima
int i=0;
rd.next(); rd.next(); rd.next(); rd.next(); // para P2, 30, 43, 255
while( rd.hasNext() )
{               tmp = rd.next();
                if(tmp.compareTo("255") == 0) pw.print ("100 ");
                else  pw.print ("255 ");
                if (++i%30==0) pw.printf ("\n");          }
rd.close();
pw.close();
}      }
```

Troca de valores 255 e 100

```
P2
30 43
255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 100 100 100 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 100 255 255 255
255 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 100 255 255 255
255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 255 255 100 255 255
255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 100 100 100 255 255 255 255 255 100 100 100 255 255 255 100 255
255 255 255 255 255 100 255 255 255 100 100 100 255 255 255 255 255 100 100 100 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 100 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 100 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 100 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 100 100 100 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 100 255
255 255 255 255 255 100 255 255 255 255 255 255 100 100 100 100 100 100 100 100 255 255 255 255 100
255 255 255 255 255 255 100 255 255 255 255 255 255 100 100 100 100 100 100 255 255 255 255 100 255
255 255 255 255 255 255 255 100 255 255 255 255 255 255 100 100 255 255 255 255 255 100 255 255 255
255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 255 100 255 255 255
255 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 255 100 255 255
255 255 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 100 255 255 255
255 255 255 255 255 255 255 255 255 255 255 100 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 100 100 100 100 100 100 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
```

## Exemplo 5:

```
import java.util.*;                                  // exemplo 5
import java.io.*;
public class ZoomFaceImage {
public static void main (String args[])  throws IOException     {
int largura, altura;
String tmp = new String();
File orid = new File("face_image.pgm");
Scanner rd = new Scanner(orid);
File fout = new File("wide_face_image.pgm");
PrintWriter pw = new PrintWriter(fout);
pw.println("P2");                    // tipo
pw.println("150 215");               // largura, altura
pw.println("255 ");                  // intensidade máxima
rd.next(); rd.next(); rd.next(); rd.next();
while( rd.hasNext() )
{            tmp = rd.next();
             if(tmp.compareTo("255") == 0) for(int a=0; a<5; a++) pw.print ("100 ");
             else  for(int a=0; a<5; a++) pw.print ("255 ");
}
rd.close();
pw.close();
}      }
```

e-learning

## Exemplo 6:

```
import java.util.*;
import java.io.*;
public class sin_cos_to_file {                                        // Exemplo 6
  static Scanner read = new Scanner(System.in);
public static void main (String args[])  throws IOException    {
  double si;
  File fout = new File("to_write_cos.pgm");
  PrintWriter pw = new PrintWriter(fout);
pw.println("P2");                          // tipo
pw.println("1256 314");                    // largura, altura
pw.println("255 ");                        // intensidade máxima
  for(int j = 628; j>=0; j--)
  for(int i = -314; i<314; i++) {si = Math.cos((double)i%314/20);
            if(((int)(si*200+300) > j-3)&&((int)(si*200+300) < j+3)) pw.print("255 ");
            else pw.print("55 ");
 }
pw.close();
}
}
```

e-learning

Exemplo 7:

```
import java.util.*;
import java.io.*;
public class sin_cos_to_file {                    // Exemplo 7
  static Scanner read = new Scanner(System.in);
public static void main (String args[])  throws IOException    {
  double si;
  File fout = new File("to_write_cos.pgm");
  PrintWriter pw = new PrintWriter(fout);
pw.println("P2");                    // tipo
pw.println("1256 314");              // largura, altura
pw.println("255 ");                  // intensidade máxima
for(int j = 628; j>=0; j--)
for(int i = -314; i<314; i++) {si = Math.sin((double)i%314/20);
            if(((int)(si*200+300) > j-3)&&((int)(si*200+300) < j+3)) pw.print("55 ");
            else pw.print("255 ");
 }
pw.close();
}
}
```

e-learning

cos

sin

# Exemplo 8:

```
import java.util.*;    import java.io.*;
public class Rotate {                                   // Exemplo 8
  static Scanner read = new Scanner(System.in);
  static final int largura = 1256, altura = 314;
public static void main (String args[])  throws IOException     {
  cos_image();
  System.out.print("Limiar (0 - 255)? ");      // tentar 150.
  threshold(read.nextInt());
  rotate();
}
public static void cos_image()   throws IOException
{   double si;
    File fout = new File("cos.pgm");
    PrintWriter pw = new PrintWriter(fout);
    pw.println("P2");
    pw.printf("%s %s\n",largura, altura);
    pw.println("255 ");
    for(int j = 628; j>=0; j--)
        for(int i = -314; i<314; i++) {si = Math.cos((double)i%314/20);
              if(((int)(si*200+300) > j-3)&&((int)(si*200+300) < j+3)) pw.print("255 ");
              else pw.print("55 ");    }
pw.close();                                             }
public static void threshold(int thr) throws IOException {
    int tmp;
    File orid = new File("cos.pgm");
    Scanner rd = new Scanner(orid);
    File fout = new File("threshold_cos.pgm");
    PrintWriter pw = new PrintWriter(fout);
pw.println("P2");
pw.printf("%s %s\n",largura, altura);
pw.println("255 ");
rd.next(); rd.next(); rd.next(); rd.next(); // para P2, 30, 43, 255
while( rd.hasNextInt() )
   {   tmp = rd.nextInt();
       pw.printf("%s ",(tmp > thr) ? 0xFF : 0);    }          }
```

```
  public static void rotate() throws IOException {
    int image[][] = new int[altura][largura];
    File orid = new File("cos.pgm");
    Scanner rd = new Scanner(orid);
    File fout = new File("rot_cos.pgm");
    PrintWriter pw = new PrintWriter(fout);
    rd.next(); rd.next(); rd.next(); rd.next();
    for(int i = 0; i < altura; i++)
       for(int j = 0; j < largura; j++)
           image[i][j] = rd.nextInt();
    pw.println("P2");
    pw.printf("%s %s\n",altura, largura);
    pw.println("255 ");
    for(int i = 0; i < largura; i++)
       for(int j = 0; j < altura; j++)
          pw.printf("%s ",image[j][i]);
  }
}
```

e-learning

Para reduzir o tamanho do código várias verificações não foram feitas (mas devem ser feitas)

Para reduzir o tamanho do código os comentários não foram removidos do ficheiro original

**USAR FICHEIROS (por exemplo, lena.pgm) SEM COMENTÁRIOS !!!**

## Exemplo 8:

### Os resultados:

```
Limiar (0 - 255)? 120
Press any key to continue . . .
```

threshold_cos.pgm 12/8/2014 7:17 PM IrfanView PGM File 768 KB

rot_cos.pgm 12/8/2014 7:17 PM IrfanView PGM File 1,144 KB

## Exemplo 9:

```
import java.util.*;     // comments cannot be removed for simplicity
import java.io.*;       // Removing comments can be found in Joao program
public class RotateLena {                                   // Exemplo 9
  static Scanner read = new Scanner(System.in); // verification of files is not done
  static final int largura = 213, altura = 213;                  // see Joao program where it is done
  static final String in = new String("Lena.pgm");   // IMPORTANT: THIS FILE CANNOT CONTAIN COMMENTS
  static final String outt = new String("thres_Lena.pgm");
  static final String outr = new String("rot_Lena.pgm");
public static void main (String args[])  throws IOException     {
  System.out.print("Limiar (0 - 255)? "); // tentar 100
  threshold(read.nextInt());     rotate();         }
public static void threshold(int thr) throws IOException {
    int tmp;  File orid = new File(in);   Scanner rd = new Scanner(orid);
    File fout = new File(outt);              PrintWriter pw = new PrintWriter(fout);
pw.println("P2");  pw.printf("%s %s\n",largura, altura);   pw.println("255 ");
rd.next(); rd.next(); rd.next(); rd.next(); // para P2, 30, 43, 255
while( rd.hasNextInt() )
{   tmp = rd.nextInt();   pw.printf("%s ",(tmp > thr) ? 0xFF : 0);    }          }
 public static void rotate() throws IOException {
    int image[][] = new int[altura][largura];
    File orid = new File(in);       Scanner rd = new Scanner(orid);
    File fout = new File(outr);    PrintWriter pw = new PrintWriter(fout);
    rd.next(); rd.next(); rd.next(); rd.next(); // para P2, 30, 43, 255
    for(int i = 0; i < altura; i++)
        for(int j = 0; j < largura; j++)     image[i][j] = rd.nextInt();
   pw.println("P2");   pw.printf("%s %s\n",altura, largura);   pw.println("255 ");
    for(int i = 0; i < largura; i++)
       for(int j = 0; j < altura; j++)   pw.printf("%s ",image[j][i]);
 }
}
```

Sem comentários

P2
213 213
255

Lena.pgm

thres_Lena.pgm

rot_Lena.pgm

e-learning

# Sumario

1. Esta aula inclui exemplos mais complicados.
2. Praticamente todos os programas estão disponíveis no elearning.ua.pt (aula 11).