

Δομές Δεδομένων & Ανάλυση Αλγορίθμων

3ο Εξάμηνο

ΑΣΚΗΣΗ ΠΡΑΞΗΣ 7

Ακολουθιακά αρχεία και ρεύματα
(Sequential files and streams)

Δημοσθένης Σταμάτης

<http://www.iee.ihu.gr/~demos>

Τμήμα Μηχανικών Πληροφορικής & Ηλεκτρονικών Συστημάτων



Ιεραρχία κλάσεων για την επεξεργασία αρχείων (package java.io)

InputStream (abstract class)

FileInputStream

FilterInputStream

DataInputStream (υλοποιεί το interface DataInput)

BufferedInputStream

LineNumberInputStream

ObjectInputStream

OutputStream (abstract class)

FileOutputStream

FilterOutputStream

DataOutputStream (υλοποιεί το DataOutput)

BufferedOutputStream

ObjectOutputStream

Δημιουργία αρχείου από bytes

```
import java.io.*;
```

```
public class WriteBytesToFile {
```

```
    public static void main(String[] arguments) {
```

```
        int[ ] data = { 71, 73, 70, 56, 57, 97, 16, 0, 15, 0, 128, 0, 0,
                        255, 255, 255, 0, 0, 0, 44, 0, 0, 0, 0, 15, 0, 15, 0, 0, 2, 33,
                        132, 127, 161, 200, 185, 205, 84, 128, 241, 81, 35, 175,
                        155, 26, 228, 254, 105, 33, 102, 121, 165, 201, 145, 169,
                        154, 142, 172, 116, 162, 240, 90, 197, 5, 0, 59 };
```

```
        try {
```

```
            FileOutputStream file = new FileOutputStream("File_with_Bytes");
```

```
            for (int i = 0; i < data.length; i++)
```

```
                file.write(data[i]);
```

```
            file.close();
```

```
        } catch (IOException error) {
            System.out.println("Error -- " + error);
        }
```

```
    }
```

```
}
```

Δημιουργία stream
εισόδου από το αρχείο

Εγγραφή ενός byte
στο αρχείο

Κλείσιμο του αρχείου

Διάβασμα bytes από αρχείο

```
import java.io.*;

public class ReadBytesFromFile {

    public static void main(String[] arguments) {
        try {
            FileInputStream file = new FileInputStream("File_with_Bytes");
            boolean eof = false;
            int count = 0;
            while (!eof) {
                int input = file.read();
                System.out.print(input + " ");
                if (input == -1) eof = true;
                else count++;
            }
            file.close();
            System.out.println("\nBytes read: " + count);
        } catch (IOException e) {
            System.out.println("Error -- " + e);
        }
    }
}
```

Δημιουργία stream
εισόδου προς το αρχείο

Είσοδος ενός byte
από το αρχείο

Τέλος του αρχείου

Κλείσιμο του αρχείου

Δημιουργία αρχείου με ακέραιους

```
import java.io.*;

public class WriteIntsToFile {

    public static void main(String arguments[]) {
        int[ ] matrix = new int[40];
        for (int i=0; i<matrix.length; i++)
            matrix[i]=i*20;

        try {
            FileOutputStream file = new FileOutputStream("File_with_Integers.dat");
            DataOutputStream data = new DataOutputStream(file);
            for (int i = 0; i<matrix.length; i++)
                data.writeInt(matrix[i]);
            data.close();
        } catch (IOException e) {
            System.out.println("Error -- " + e.toString());
        }
    }
}
```

Δημιουργία αρχείου με ακέραιους (με χρήση buffer)

```
import java.io.*;

public class WriteIntsToFile {

    public static void main(String arguments[]) {
        int[ ] matrix = new int[40];
        for (int i=0; i<matrix.length; i++)
            matrix[i]=i*20;

        try {
            FileOutputStream file = new FileOutputStream("File_with_Integers.dat");
            BufferedOutputStream buff = new BufferedOutputStream(file);
            DataOutputStream data = new DataOutputStream(buff);
            for (int i = 0; i<matrix.length; i++)
                data.writeInt(matrix[i]);
            data.close();
        } catch (IOException e) {
            System.out.println("Error -- " + e.toString());
        }
    }
}
```

Διάβασμα ακεραίων από αρχείο

```
import java.io.*;

public class ReadIntsFromFile {
    public static void main(String arguments[]) {
        try {
            FileInputStream file = new FileInputStream("File_with_Integers");
            BufferedInputStream buff = new BufferedInputStream(file);
            DataInputStream data = new DataInputStream(buff);
            try {
                while (true) {
                    int in = data.readInt();
                    System.out.print(in + " ");
                }
            } catch (EOFException eof) {
                buff.close();
            }
        } catch (IOException e) {
            System.out.println("Error -- " + e.toString());
        }
    }
}
```

Συγχώνευση δύο ταξινομημένων αρχείων (1/4)

```
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.EOFException;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;

public class MergeFiles {

    public static void main(String arguments[]) {
        try {
            FileInputStream file1 = new FileInputStream("File_with_Integers1");
            DataInputStream data1 = new DataInputStream(file1);
            FileInputStream file2 = new FileInputStream("File_with_Integers2");
            DataInputStream data2 = new DataInputStream(file2);
            FileOutputStream file3 = new FileOutputStream("File_with_Integers3");
            DataOutputStream data3 = new DataOutputStream(file3);

            int input_from_file1=0;
            int input_from_file2=0;
```


Συγχώνευση δύο ταξινομημένων αρχείων (2/4)

```
try {  
    input_from_file1 = data1.readInt();  
    input_from_file2 = data2.readInt();  
    while (true) {  
        if (input_from_file1 < input_from_file2){  
            data3.writeInt(input_from_file1);  
            System.out.println(input_from_file1+" from file1 ");  
            input_from_file1 = data1.readInt(); }  
        else {  
            data3.writeInt(input_from_file2);  
            System.out.println(input_from_file2+" from file2 ");  
            input_from_file2 = data2.readInt(); }  
        }  
    }  
catch (EOFException eof) {
```

Συγχώνευση δύο ταξινομημένων αρχείων (3/4)

```
catch (EOFException eof) {  
    if (data1.available() > 0) { // if the first file has elements left  
        data2.close();           // the second file is empty  
        System.out.println(" exit file2 ");  
        try {  
            while (true) {  
                data3.writeInt(input_from_file1);  
                System.out.println(input_from_file1 + " from file1 ");  
                input_from_file1 = data1.readInt(); }  
            }  
        catch (EOFException eof1) {  
            data1.close();  
            data3.close();  
        }  
    }  
    else { //if the second file has elements left
```

Συγχώνευση δύο ταξινομημένων αρχείων (4/4)

```
else {                                     //if the second file has elements left
    data1.close();                         // the first file is empty
    System.out.println(" exit file1 ");
    try {
        while (true){
            data3.writeInt(input_from_file2);
            System.out.println(input_from_file2+" from file2 ");
            input_from_file2 = data2.readInt(); }
        }
    catch (EOFException eof1) {
        data2.close();
        data3.close();
    }
}

}

catch (IOException e) {
    System.out.println("Error -- " + e.toString());
}
```