

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

Δομές Δεδομένων & Ανάλυση Αλγορίθμων

3ο Εξάμηνο

• Αρχεία και Ρεύματα Bytes

(Files an Byte Streams)

Δημοσθένης Σταμάτης

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

Τμήμα Μηχανικών Πληροφορικής & Ηλεκτρονικών Συστημάτων

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

ΑΡΧΕΙΑ (FILES)

➤ Ένα πρόγραμμα αποθηκεύει τα δεδομένα που χρησιμοποιεί και παράγει σε μεταβλητές.

➤ Όμως όταν τερματιστεί η εκτέλεση του προγράμματος οι μεταβλητές χάνονται - αποδεσμεύεται η μνήμη που χρησιμοποιεί το πρόγραμμα.

➤ Πως μπορεί ένα πρόγραμμα να σώσει δεδομένα έτσι ώστε να μην χαθούν, ακόμα και μετά τον τερματισμό του, π.χ. για να τα χρησιμοποιήσει την επόμενη φορά που θα εκτελεστεί;

➤ Η λύση στο πρόβλημα: **Αρχεία!**

➤ Ένα αρχείο δεν είναι παρά μια ακολουθία από bytes αποθηκευμένα στον δίσκο του Η/Υ (και όχι στην μνήμη όπως οι πίνακες).

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

ΑΡΧΕΙΑ (FILES)

👉 Ένα **αρχείο (file)** είναι μία σύνθετη δομή δεδομένων που αποτελείται από μία σειρά στοιχείων (**elements**), τα οποία είναι (συνήθως) του ίδιου τύπου και είναι αποθηκευμένα σε ένα χώρο αποθήκευσης (σκληρό δίσκο, usb κλπ.).

👉 **Ανάλογα με τον τρόπο αποθήκευσης:**

➤ **Αρχεία κειμένου (text files)**

➤ **Διαδικά αρχεία (binary files)**

👉 **Ανάλογα με τον τρόπο προσπέλασης:**

➤ **Σειριακά ή ακολουθιακά αρχεία (sequential files)**

➤ **Αρχεία κατ' ευθείαν πρόσβασης (direct access)**

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

Ρεύματα (Streams)

👉 **Ρεύμα (Stream)** είναι ένα αντικείμενο το οποίο αναπαριστά μία σειριακή ροή δεδομένων από μία πηγή προς έναν προορισμό.

👉 Η Java αντιμετωπίζει ένα αρχείο σαν ένα ακολουθιακό ρεύμα (stream) από bytes. Κάθε αρχείο είτε τελειώνει με το ειδικό σύμβολο "end-of-file" είτε με ένα byte που περιέχει έναν συγκεκριμένο αριθμό που καθορίζεται από το σύστημα.

👉 Ένα ρεύμα μπορεί να οριστεί ως μια ακολουθία δεδομένων. Υπάρχουν δύο είδη ρευμάτων:

➤ **InputStream** - Χρησιμοποιείται για την ανάγνωση δεδομένων από μία πηγή.

➤ **OutputStream** - Χρησιμοποιείται για την εγγραφή δεδομένων σε κάποιο προορισμό.

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

Class Hierarchy

class java.lang.Object

class java.io.File (implements java.lang.Comparable, java.io.Serializable)

class java.io.FileDescriptor

class java.io.InputStream

class java.io.ByteArrayInputStream

class java.io.FilterInputStream

class java.io.BufferedInputStream

class java.io.DataInputStream (implements java.io.DataInput)

class java.io.LineNumberInputStream

class java.io.PushbackInputStream

class java.io.ObjectInputStream (implements java.io.ObjectInput, java.io.ObjectStreamConstants)

class java.io.PipedInputStream

class java.io.SequenceInputStream

class java.io.StringBufferInputStream

class java.io.ObjectInputStream.GetField

class java.io.ObjectOutputStream.PutField

class java.io.ObjectStreamClass (implements java.io.Serializable)

class java.io.ObjectStreamField (implements java.lang.Comparable)

class java.io.OutputStream

class java.io.ByteArrayOutputStream

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

Βασικές κλάσεις ρευμάτων

InputStream

SequenceInputStream

PipedInputStream

FilteredInputStream

StringBufferInputStream

FileInputStream

ByteArrayInputStream

DataInputStream

PushbackInputStream

BufferedInputStream

LineNumberInputStream

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

1

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΤΑΕ

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

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΤΑΕ

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

Reader (abstract class)
BufferedReader
InputStreamReader
 FileReader

Writer (abstract class)
BufferedWriter
OutputStreamWriter
 FileWriter

RandomAccessFile (υλοποιεί τα interfaces DataInput & DataOutput)

File

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΤΑΕ

Ρεύμα Εισόδου (Input Stream)

Είσοδος δεδομένων από ένα εξωτερικό μέσο
(π.χ. αρχείο στο δίσκο) προς το πρόγραμμα

The diagram illustrates an input stream. On the left, a cylinder represents a 'Disk File' (Αρχείο στο δίσκο). An arrow points from this cylinder to a central pipe. Inside the pipe, several blue arrows point from left to right, representing the 'Flow of Data' (Ροή Δεδομένων). The pipe ends on the right with an arrow pointing into a box representing the 'Program' (Πρόγραμμα).

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΤΑΕ

Ρεύμα Εξόδου (Output Stream)

Έξοδος δεδομένων από το πρόγραμμα προς ένα
εξωτερικό μέσο (π.χ. αρχείο στο δίσκο)

The diagram illustrates an output stream. On the right, a box represents the 'Program' (Πρόγραμμα). An arrow points from this box to a central pipe. Inside the pipe, several blue arrows point from right to left, representing the 'Flow of Data' (Ροή Δεδομένων). The pipe ends on the left with an arrow pointing into a cylinder representing a 'Disk File' (Αρχείο στο δίσκο).

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΤΑΕ

Ρεύμα Εισόδου (input stream): Διαδικασία Ανάγνωσης

ΒΗΜΑ 1: Δημιούργησε ένα ρεύμα εισόδου με βάση το αρχείο;

ΒΗΜΑ 2: Εφόσον <υπάρχουν δεδομένα>
 Διάβασε επόμενο δεδομένο;
 [Επεξεργάσου το δεδομένο;]

ΒΗΜΑ 3: Κλείσε το ρεύμα εισόδου;

Ρεύμα Εξόδου (output stream): Διαδικασία Εγγραφής

ΒΗΜΑ 1: Δημιούργησε ένα ρεύμα εξόδου με βάση το αρχείο;

ΒΗΜΑ 2: Εφόσον <υπάρχουν δεδομένα>
 [Προετοίμασε επόμενο δεδομένο;]
 Γράψε δεδομένο;

ΒΗΜΑ 3: Κλείσε το ρεύμα εξόδου;

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΤΑΕ

Ρεύμα Εισόδου (input stream): Διαδικασία Ανάγνωσης

ΒΗΜΑ 1: Δημιούργησε ένα ρεύμα εισόδου με βάση το αρχείο;

ΒΗΜΑ 2: Εφόσον <υπάρχουν δεδομένα>
 Διάβασε επόμενο δεδομένο;
 [Επεξεργάσου το δεδομένο;]

ΒΗΜΑ 3: Κλείσε το ρεύμα εισόδου;

Ρεύμα Εξόδου (output stream): Διαδικασία Εγγραφής

ΒΗΜΑ 1: Δημιούργησε ένα ρεύμα εξόδου με βάση το αρχείο;

ΒΗΜΑ 2: Εφόσον <υπάρχουν δεδομένα>
 [Προετοίμασε επόμενο δεδομένο;]
 Γράψε δεδομένο;

ΒΗΜΑ 3: Κλείσε το ρεύμα εξόδου;

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

Βασικές Μέθοδοι της κλάσης `InputStream`

Μέθοδος	Περιγραφή
<code>int read()</code>	Επιστρέφει έναν ακέραιο αριθμό που αντιστοιχεί στο επόμενο διαθέσιμο byte του stream εισόδου. Επιστρέφει -1 όταν συναντήσει το τέλος του αρχείου.
<code>int read(byte buffer[])</code>	Προσπαθεί να διαβάσει από το stream εισόδου αριθμό από bytes ίσο με το μήκος <code>buffer.length</code> του πίνακα <code>buffer</code> και τους τοποθετεί σε αυτόν. Επιστρέφει έναν ακέραιο αριθμό που αντιστοιχεί στον αριθμό των bytes που διάβασε ή -1 αν συναντήσει το τέλος του αρχείου.
<code>int available()</code>	Επιστρέφει τον αριθμό των bytes του stream εισόδου που είναι διαθέσιμα.
<code>long skip(long numBytes)</code>	Προσπαθεί να παρακάμψει τα επόμενα <code>numBytes</code> bytes του stream εισόδου επιστρέφοντας τον αριθμό όσων αγνόησε.
<code>void close()</code>	Κλείνει το stream εισόδου. Μετά το κλείσιμο προσπάθειες ανάγνωσης δεδομένων οδηγούν σε εξαίρεση <code>IOException</code>

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

Βασικές Μέθοδοι της κλάσης `OutputStream`

Μέθοδος	Περιγραφή
<code>void write(int oneByte)</code>	Γράφει ένα μόνο byte στο stream εξόδου.
<code>void write(byte buffer[])</code>	Γράφει τα bytes που είναι αποθηκευμένα στον πίνακα <code>buffer</code> στο stream εξόδου.
<code>void close()</code>	Κλείνει το stream εξόδου. Μετά το κλείσιμο προσπάθειες εγγραφής δεδομένων οδηγούν σε εξαίρεση <code>IOException</code>

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

Περιγραφή της `read` της `FileInputStream`

```
public int read(byte[ ] b) throws IOException
```

Διαβάζει έως `b.length` bytes δεδομένων από το ρεύμα εισόδου (input stream) και τους τοποθετεί σε έναν πίνακα από bytes. Η μέθοδος αυτή μπαίνει σε κατάσταση αναμονής (blocks) μέχρις ότου υπάρχουν διαθέσιμα δεδομένα.

Παράμετροι (Parameters):
b – ο πίνακας στον οποίο τοποθετούνται τα δεδομένα που διαβάζονται.

Επιστρέφει (Returns):
Τον ολικό αριθμό των Bytes που διαβάστηκαν στον πίνακα ή -1 αν δεν υπάρχουν άλλα δεδομένα γιατί η διαδικασία ανάγνωσης έφτασε στο τέλος του αρχείου (end of the file).

Προκαλεί (Throws):
`IOException` – εάν προκύψει λάθος εισόδου/εξόδου (I/O error).

Επιαναορίζει (Overrides):
Τη `read` της κλάσης `InputStream`

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

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, 44, 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 εξόδου στο αρχείο

Εγγραφή ενός byte στο αρχείο

Κλείσιμο του αρχείου

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

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.println(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 από το αρχείο

Εισόδος ενός byte από το αρχείο

Τέλος του αρχείου

Κλείσιμο του αρχείου

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΤΠΔΕ

Ενδεικτικές Μέθοδοι της κλάσης `DataInputStream`

Μέθοδος	Περιγραφή
<code>boolean readBoolean()</code>	Διαβάζει το επόμενο byte από το ρεύμα εισόδου. Επιστρέφει true εάν η τιμή του είναι μη μηδενική και false εάν είναι μηδέν.
<code>int readInt()</code>	Διαβάζει τα επόμενα τέσσερα bytes από το ρεύμα εισόδου και επιστρέφει τον ακέραιο αριθμό τύπου int που αντιστοιχεί σε αυτά.
<code>float readFloat()</code>	Διαβάζει τα επόμενα τέσσερα bytes από το ρεύμα εισόδου και επιστρέφει τον πραγματικό αριθμό τύπου float που αντιστοιχεί σε αυτά.

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝΔΙΠΤΑΕ

Ενδεικτικές Μέθοδοι της κλάσης `DataOutputStream`

Μέθοδος	Περιγραφή
<code>void writeInt(int n)</code>	Γράφει έναν ακέραιο αριθμό στο ρεύμα εξόδου σαν μία σειρά από τέσσερα bytes, αρχίζοντας από το περισσότερο σημαντικό byte.
<code>void writeFloat(float n)</code>	Μετατρέπει τον πραγματικό αριθμό n σε ακέραιο χρησιμοποιώντας τη μέθοδο <code>floatToIntBits</code> της κλάσης <code>Float</code> , και στη συνέχεια γράφει τον αριθμό αυτό στο ρεύμα εξόδου σαν μία σειρά από τέσσερα bytes, αρχίζοντας από το περισσότερο σημαντικό.
<code>void close()</code>	Κλείνει το stream εξόδου. Μετά το κλείσιμο προσπάθειες εγγραφής δεδομένων οδηγούν σε εξαίρεση <code>IOException</code>

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝΓ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝΔΙΠΤΑΕ

```
import java.io.*;
public class ReadsFromFile {
    public static void main(String arguments[]) {
        try {
            FileInputStream file = new FileInputStream("File_with_Integers.dat");
            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());
        }
    }
}
```

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝΓ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝΔΙΠΤΑΕ

```
import java.io.*;

public class WritesToFile {

    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());
        }
    }
}
```

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝΓ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝΔΙΠΤΑΕ

```
import java.io.*;

public class WritesToFile {

    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());
        }
    }
}
```

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝΓ' ΕΞΑΜΗΝΟ

ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝΔΙΠΤΑΕ

```
import java.io.*;

public class CopyFile {

    public static void main(String args[]) throws IOException {
        FileInputStream in = null;
        FileOutputStream out = null;

        try {
            in = new FileInputStream("input.dat");
            out = new FileOutputStream("output.txt");
            int c;
            while ((c = in.read()) != -1) {
                out.write(c);
            }
        } finally {
            if (in != null) {
                in.close();
            }
            if (out != null) {
                out.close();
            }
        }
    }
}
```

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝΓ' ΕΞΑΜΗΝΟ