

Lezione 10: I File parte seconda

Laboratorio di Elementi di Architettura e Sistemi Operativi

10 Aprile 2013

Esempio 1

Scrivere un programma C che:

- prende come *parametri della linea di comando* due nomi di file;
- copia il contenuto del primo file nel secondo (riscrivendolo se esiste già).

Prima soluzione: lettura e scrittura a caratteri

```
#include <stdio.h>
#include <stdlib.h>

void printerror(char *str, int n) {
    fprintf(stderr, "Errore %d: %s\n", n, str);
    exit(n);
}

int main(int argc, char *argv[]) {
    FILE *fin, *fout;
    int c, res;

    if (argc != 3)
        printerror("Specificare due file come argomenti", 1);
    fin = fopen(argv[1], "r");
    if (fin == NULL)
        printerror("Errore nell'apertura del primo file.", 2);
    fout = fopen(argv[2], "w");
    if (fout == NULL)
        printerror("Errore nell'apertura del secondo file.", 2);

    c = fgetc(fin);
    while (c != EOF) {
        res = fputc(c, fout);
        if (res == EOF) {
            printerror("Errore nella scrittura del secondo file.", 3);
        }
        c = fgetc(fin);
    }

    if (fclose(fin) != 0) {
        printerror("Errore nella chiusura del primo file.", 4);
    }
    if (fclose(fout) != 0) {
        printerror("Errore nella chiusura del secondo file.", 4);
    }
    return 0;
}
```

Seconda soluzione: lettura e scrittura per righe

```
#include <stdio.h>
#include <stdlib.h>

void printerror(char *str, int n) {
    fprintf(stderr, "Errore %d: %s\n", n, str);
    exit(n);
}

int main(int argc, char *argv[]) {
    FILE *fin, *fout;
    char s[100];
    char *res;

    if (argc != 3)
        printerror("Specificare due file come argomenti", 1);

    fin = fopen(argv[1], "r");
    if (fin == NULL)
        printerror("Errore nell'apertura del primo file.", 2);

    fout = fopen(argv[2], "w");
    if (fout == NULL)
        printerror("Errore nell'apertura del secondo file.", 2);

    res = fgets(s, 100, fin);
    while (res != NULL) {
        if (fputs(s, fout) == EOF) {
            printerror("Errore nella scrittura del secondo file.", 3);
        }
        res = fgets(s, 100, fin);
    }

    if (fclose(fin) != 0) {
        printerror("Errore nella chiusura del primo file.", 4);
    }
    if (fclose(fout) != 0) {
        printerror("Errore nella chiusura del secondo file.", 4);
    }
    return 0;
}
```

Esempio 2

Scrivere un programma C che:

- prende come *parametri della linea di comando* due nomi di file;
- legge una sequenza di numeri dal primo file, rappresentati *in decimale*;
- copia la sequenza nel secondo file, scrivendo i numeri *in esadecimale*.

Soluzione: lettura e scrittura formattata

```
#include <stdio.h>
#include <stdlib.h>

void printerror(char *str, int n) {
    fprintf(stderr, "Errore %d: %s\n", n, str);
    exit(n);
}

int main(int argc, char *argv[]) {
    FILE *fin, *fout;
    int res, n;

    if (argc != 3)
        printerror("Specificare due file come argomenti", 1);
    fin = fopen(argv[1], "r");
    if (fin == NULL)
        printerror("Errore nell'apertura del primo file.", 2);
    fout = fopen(argv[2], "w");
    if (fout == NULL)
        printerror("Errore nell'apertura del secondo file.", 2);

    res = fscanf(fin, "%d", &n);
    while (res != EOF) {
        if (res != 1) {
            printerror("Input in formato non corretto.", 3);
        }
        if (fprintf(fout, "%x\n", n) == EOF) {
            printerror("Errore nella scrittura del file.", 4);
        }
        res = fscanf(fin, "%d", &n);
    }

    if (fclose(fin) != 0) {
        printerror("Errore nella chiusura del primo file.", 5);
    }
    if (fclose(fout) != 0) {
        printerror("Errore nella chiusura del secondo file.", 5);
    }
    return 0;
}
```