



Programmazione 1

Accesso File in C

Michele Nappi
mnappi@unisa.it

Il File: Dichiarazione

- In C è definito un tipo **FILE**

FILE *infile

FILE *outfile

- Le variabili **infile** e **outfile** sono puntatori a file

Il File: Accesso

- La dichiarazione di un file non è associata alla creazione del file
- Utilizzo della funzione *fopen*

file pointer variable=fopen(*file name*, *mode*);

- *file name*: stringa che specifica il nome del file
- *mode*: stringa che specifica la modalità di trasferimento dati
 - “r”: lettura (il file deve già esistere)
 - “w”: scrittura (viene creato un file di nome *file name*)
 - “a”: *appending*

Il File: Accesso

- Possibili modalità di trasferimento
 - “rw”: lettura e scrittura (“r+”)
 - “wr”: scrittura e lettura (“w+”)
 - “ra”: lettura e appending
 - “b”: *apertura di file binario*
- Lettura e Scrittura sullo stesso File (o viceversa)
 - fflush(FILE *fp)
 - Svuota il buffer
 - fseek(FILE *fp, long offset, int place)
 - La posizione è **offset** byte da **place**
 - rewind(FILE *fp)
 - La posizione è all’inizio del File (void fseek(fp,0L,Seek_Set))

Il File: Chiusura

- I file aperti in C devono essere chiusi al termine dell'esecuzione
 - *fclose(infile)*
 - *fclose(outfile)*

Il File: Scrittura e Lettura

- `getc(pointer to file)`
- `putc(char,pointer to file)`
- `fscanf(pointer,.....)`
- `fprintf(pointer,.....)`

Esempio: Scrivere un programma in C che presi in input due file, denominati *fileuno* e *filedue* contenenti caratteri minuscoli, copi il contenuto di *filedue* in coda a *fileuno*, e successivamente crei un nuovo file, il cui nome è inserito dall'utente, in cui sia copiato il contenuto di *fileuno* invertendo le minuscole con le maiuscole.

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

```
/*definizione prototipi*/
```

```
int controllo_input(int);  
void controllo_file(FILE *, char *);  
int append_file(FILE *, FILE *);  
int converti_maiuscole(char *);
```

```
int main(int argc, char *argv[])
```

```
{
```

```
    FILE *uno, *due;
```

```
    controllo_input(argc);
```

```
    uno=fopen(argv[1],"a");
```

```
    controllo_file(uno,argv[1]);
```

```
    due=fopen(argv[2],"r");
```

```
    controllo_file(due,argv[2]);
```

```
    if(append_file(uno,due))
```

```
        printf("La scrittura è OK\n");
```

```
    else
```

```
        printf("La scrittura non è
```

```
        avvenuta\n");
```

```
    if(converti_maiuscole(argv[1])
```

```
        printf("La conversione è OK\n");
```

```
    return 0;
```

```
}
```

Esempio (continua)

```
int controllo_input (int n)

{
    if(n!=3)
    {
        printf("Il programma apre due file denominati
        fileuno e filedue\n");
        printf("preceduti dal nome programma: il
        risultato sarà la concatenazione\n");
        printf("dei due file memorizzata in fileuno.
        Successivamente il contenuto\n");
        printf("di fileuno verrà copiato in un file
        nuovo, il cui nome verrà deciso\n");
        printf("dall'utente, convertendo le minuscole
        in maiuscole\n");
        exit(1);
    }
    else
    return 0;
}
```

```
#include <stdio.h>

/*definizione prototipi*/

int controllo_input(int);
void controllo_file(FILE *, char *);
int append_file(FILE *, FILE *);
int converti_maiuscole(char *);

int main(int argc, char *argv[])
{
    FILE *uno, *due;

    controllo_input(argc);
    uno=fopen(argv[1],"a");
    controllo_file(uno,argv[1]);
    due=fopen(argv[2],"r");
    controllo_file(due,argv[2]);
    if(append_file(uno,due))
        printf("La scrittura è OK\n");
    else
        printf("La scrittura non è avvenuta\n");
    if(converti_maiuscole(argv[1]))
        printf("La conversione è OK\n");
    return 0;
}
```


Esempio (continua)

```
void controllo_file(FILE *file, char
*stringa)

{
if(file!=NULL)
printf("%s è stato aperto
correttamente\n",
stringa);
else
exit(1);
}
```

```
#include <stdio.h>

/*definizione prototipi*/

int controllo_input(int);
void controllo_file(FILE *, char *);
int append_file(FILE *, FILE *);
int converti_maiuscole(char *);

int main(int argc, char *argv[])
{
    FILE *uno, *due;

    controllo_input(argc);
    uno=fopen(argv[1],"a");
    controllo_file(uno,argv[1]);
    due=fopen(argv[2],"r");
    controllo_file(due,argv[2]);
    if(append_file(uno,due))
        printf("La scrittura è OK\n");
    else
        printf("La scrittura non è avvenuta\n");
    if(converti_maiuscole(argv[1]))
        printf("La conversione è OK\n");
    return 0;
}
```

Esempio (continua)

```
int append_file(FILE *f1, FILE *f2)
```

```
{
    int c;
    while((c=getc(f2))!=EOF)
        putc(c,f1);
    fclose(f1);
    fclose(f2);
    return(1);
}
```

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

```
/*definizione prototipi*/
```

```
int controllo_input(int);
void controllo_file(FILE *, char *);
int append_file(FILE *, FILE *);
int converti_maiuscole(char *);
```

```
int main(int argc, char *argv[])
{
    FILE *uno, *due;

    controllo_input(argc);
    uno=fopen(argv[1],"a");
    controllo_file(uno,argv[1]);
    due=fopen(argv[2],"r");
    controllo_file(due,argv[2]);
    if(append_file(uno,due))
        printf("La scrittura è OK\n");
    else
        printf("La scrittura non è avvenuta\n");
    if(converti_maiuscole(argv[1]))
        printf("La conversione è OK\n");
    return 0;
}
```

Esempio (continua)

```
int converti_maiuscole (char *stringa)
{
    char nome[10]; int c;
    FILE *tre, *f1;
    printf("Inserisci il nome file\n");
    scanf("%s",nome);
    tre=fopen(nome,"wb");
    controllo_file(tre,nome);
    f1=fopen(stringa,"r");
    controllo_file(f1,stringa);
    while((c=getc(f1))!=EOF)
    if(c!=' ')
        putc(c-32,tre);
    else
        putc(c,tre);
    fclose(tre);
    fclose(f1);
    return (1);
}
```

```
#include <stdio.h>

/*definizione prototipi*/

int controllo_input(int);
void controllo_file(FILE *, char *);
int append_file(FILE *, FILE *);
int converti_maiuscole(char *);

int main(int argc, char *argv[])
{
    FILE *uno, *due;

    controllo_input(argc);
    uno=fopen(argv[1],"a");
    controllo_file(uno,argv[1]);
    due=fopen(argv[2],"r");
    controllo_file(due,argv[2]);
    if(append_file(uno,due))
        printf("La scrittura è OK\n");
    else
        printf("La scrittura non è avvenuta\n");
    if(converti_maiuscole(argv[1]))
        printf("La conversione è OK\n");
    return 0;
}
```