

Σημειώσεις για τα Project στα Λειτουργικά Συστήματα I

Αρχεία :

Για να διαχειριστούμε αρχεία σε προγράμματα C ή C++ πρέπει αρχικά να δηλώσουμε το κάθε αρχείο ως pointer του τύπου FILE δηλαδή :

```
FILE * file1, file2;
```

Τα ονόματα file1, file2 δεν έχουν καμία σχέση με τα πραγματικά ονόματα των αρχείων στο δίσκο. Απλά είναι μεταβλητές της C. Στη συνέχεια για να ανοίξουμε ένα αρχείο πρέπει να καλέσουμε τη συνάρτηση fopen :

```
file1=fopen("Όνομα αρχείου στο δίσκο", "τρόπος ανοίγματος");
```

Όπου "τρόπος ανοίγματος" μπορεί να είναι ένα από τα παρακάτω :

"r" : άνοιγμα για ανάγνωση

"w" : άνοιγμα για δημιουργία/γράψιμο

"a" : άνοιγμα για πρόσθεση στο τέλος του αρχείου (append)

Στα παραπάνω αν προσθέσουμε και το χαρακτήρα "+" τότε επιτρέπεται να κάνουμε όλες τις εργασίες με το αρχείο.

Επίσης αν προσθέσουμε τον χαρακτήρα "b" τότε το αρχείο ανοίγει ως «δυναμικό» αρχείο (binary). Π.χ. :

```
file1=fopen("text1.txt", "wb+");
```

ανοίγει το αρχείο "text1.txt" για γράψιμο αλλά και άλλες λειτουργίες και μάλιστα ως δυαδικό αρχείο.

Για να κλείσουμε ένα αρχείο καλούμε την συνάρτηση fclose() :

```
fclose(file1);
```

Για να διαβάσουμε και να γράψουμε σε ένα αρχείο κειμένου χρησιμοποιούμε τις συναρτήσεις fscanf και fprintf :

```
fscanf( FILE * fp, "format string", pointer list);
```

```
fprintf( FILE * fp, "format string", variable list);
```

Παράδειγμα :

```
int a;
```

```
fscanf(file1, "%i", &a);
```

διαβάζει από το αρχείο έναν ακέραιο

```
fprintf(file2, "%i", a);
```

γράφει τον ακέραιο σε άλλο αρχείο

Επίσης μπορούμε να χρησιμοποιήσουμε τις συναρτήσεις :

```
int fgetc(FILE *stream);
```

διαβάζει ένα χαρακτήρα

```
int fputc(int c, FILE *stream);
```

γράφει ένα χαρακτήρα

```
char *fgets(char *s, int n, FILE *stream);
```

διαβάζει μία γραμμή (string)

```
int fputs(const char *s, FILE *stream);
```

γράφει μία γραμμή (string)

Για να διαβάσουμε από ένα δυαδικό αρχείο καλούμε την συνάρτηση fread() :

```
fread(δείκτης σε μεταβλητή, μέγεθος, πλήθος, αρχείο)
```

Π.χ

```
int a;
```

```
fread(&a,sizeof(int),1,file2);
```

διαβάζει από το αρχείο έναν ακέραιο (2 bytes)

Για να γράψουμε σε ένα δυαδικό αρχείο καλούμε την συνάρτηση fwrite() :

```
fwrite(δείκτης σε μεταβλητή, μέγεθος, πλήθος, αρχείο)
```

Π.χ

```
int a=5;
fwrite(&a,sizeof(int),1,file1);
```

 γραφει τον ακέραιο «5» στο αρχείο (2 bytes)

Για να ανιχνεύσουμε πότε τελειώνει ένα αρχείο καλούμε την συνάρτηση feof() ;

```
int feof(FILE *stream);
```

Παράδειγμα :

```
char c;
FILE * fp;
fp=fopen("test.fdat", "r");
if (fp==NULL) printf("Error opening file");
else
{
    while (! feof(fp) )
    {
        c=fgetc(fp);
        printf("%c %i\n",c,c);
    }
    fclose(fp);
}
```

 ο βρόχος διαβάζει από το αρχείο έναν-έναν χαρακτήρα και τους τυπώνει στην οθόνη και σαν χαρακτήρες αλλά και σαν αριθμούς ASCII μέχρι να τελειώσει το αρχείο.

Πέρασμα παραμέτρων σε πρόγραμμα C

Για να περάσουμε παραμέτρους σε ένα πρόγραμμα C κατά την εκτέλεσή του στο Command Line αρκεί να δηλώσουμε την main() ως εξής :

```
void main(int argc, char ** argv)
{
}
```

όπου argc είναι ο αριθμός των παραμέτρων και argv ένας πίνακας με strings που έχει argc θέσεις. Παράδειγμα :

```
void main(int argc, char ** argv)
{
    int i;
    for (i=0 ; i<argc ; i++)
        printf("%s\n",argv[i]);
    getch();
}
```

 ο βρόχος εμφανίζει όλες τις παραμέτρους την μία κάτω από την άλλη

Διάβασμα των αρχείων ενός directory

Για να διαβάσουμε τα αρχεία ενός καταλόγου πρέπει αρχικά να δηλώσουμε μία δομή (struct) ως εξής :

```
struct fblk {
```

```

long          ff_reserved;
long          ff_fsize; /* file size */
unsigned long ff_attrib; /* attribute found */
unsigned short ff_ftime; /* file time */
unsigned short ff_fdate; /* file date */
char          ff_name[256]; /* found file name */
};

```

Για να ξεκινήσουμε το διάβασμα των ονομάτων των αρχείων του καταλόγου καλούμε την συνάρτηση findfirst :

```
int findfirst(const char *pathname, struct fblk *fblk, int attrib);
```

Παράδειγμα :

```

struct fblk fileblock;
int result;
result= findfirst("C:\\Temp\\*.*", &fileblock, 0);
if (result==0) printf("%s\n",fileblock.ff_name);   Εμφανίζει το όνομα του αρχείου που
else printf("No files\n");                       βρέθηκε αλλιώς μήνυμα λάθους

```

Αν βρεθούν αρχεία το αποτέλεσμα (result) είναι 0 αλλιώς είναι -1.

Για να συνεχίσουμε το διάβασμα των αρχείων καλούμε την findnext();

```
int findnext(struct fblk *fblk );
```

Παράδειγμα :

```

struct fblk fileblock;
int result;
result= findfirst("C:\\Temp\\*.*", &fileblock, 0);
while (result==0)
    {
        printf("%s\n",fileblock.ff_name);
        findnext(&fileblock);
    }
printf("No more files\n");

```

ο βρόχος εμφανίζει όλα τα αρχεία που βρίσκονται στο C:\Temp και όταν τελειώσουν εμφανίζει το "No more files"

Σπύρος Καζαργής