

# Αρχεία κειμένου

Ιωάννης Γ. Τσούλος

2020

## 1 Αρχεία κειμένου

Στον προγραμματισμό υπάρχουν δύο ειδών αρχεία: τα αρχεία κειμένα και τα δυαδικά αρχεία. Τα αρχεία κειμένου περιέχουν δεδομένα τα οποία μπορούν να διαβαστούν από οποιονδήποτε κειμενογράφο. Σε αυτήν την κατηγορία είναι τα αρχεία κώδικα για παράδειγμα. Στην δεύτερη κατηγορία είναι αρχεία τα οποία θέλουν ειδική εφαρμογή για την ανάγνωσή τους, όπως τα αρχεία word, τα αρχεία excel κτλ.

Τα αρχεία στην γλώσσα C ανοίγουν με την χρήση της συνάρτησης `fopen(filepath,mode)` Το πρώτο όρισμα είναι το όνομα του αρχείου που πρέπει να ανοίξει ή ακόμα και το πλήρες μονοπάτι στο σύστημα αρχείων π.χ `/home/user/test1.txt` Το δεύτερο όρισμα είναι ο τρόπος με τον οποίο ανοίγει το αρχείο. Αποδεκτές τιμές για την παράμετρο `mode` παρουσιάζονται στον επόμενο πίνακα:

Mode	Σημασία
"r"	Άνοιγμα του αρχείου για ανάγνωση
"w"	Άνοιγμα για εγγραφή. Αν το αρχείο δεν υπάρχει δημιουργείται. Αν το αρχείο δεν υπάρχει δημιουργείται.
"a"	Ανοίγει το αρχείο για εγγραφή στο τέλος. Αν το αρχείο δεν υπάρχει δημιουργείται. Αν υπάρχει τα δεδομένα, τότε αποθηκεύονται στο τέλος του
"r+"	Ανοίγει το αρχείο για ανάγνωση και εγγραφή.
"w+"	Ανοίγει το αρχείο για ανάγνωση και εγγραφή. Αν το αρχείο δεν υπάρχει δημιουργείται. Αν υπάρχει διαγράφεται
"a+"	Ανοίγει το αρχείο για ανάγνωση και προσάρτηση. Αν δεν υπάρχει δημιουργείται, αλλιώς τα δεδομένα προστιθενται στο τέλος.

Η συνάρτηση `fopen` επιστρέφει την ειδική τιμή `NULL` αν δεν μπορεί να ανοίξει το συγκεκριμένο αρχείο με τον συγκεκριμένο προσδιοριστή. Αν τα καταφέρει επιστρέφει μια ειδική δομή `FILE *` (δείκτης σε δομή `FILE` που θα εξετάσουμε και αργότερα). Το παράδειγμα του αλγορίθμου 1 προσπαθεί να ανοίξει το αρχείο `test.txt` στον τοπικό φάκελο για ανάγνωση. Αν τα καταφέρει εμφανίζει ένα σχετικό μήνυμα αλλιώς ένα μήνυμα λάθους.

---

**Algorithm 1** Προσπάθεια ανοίγματος αρχείου για ανάγνωση.

---

```
1 # include <stdio.h>
2 int main ()
3 {
4     FILE *fp=fopen (" test . txt " , " r " ) ;
5     if (fp==NULL)
6         printf ("Lathos _ sto _ anoigma _ tou _ arxeiou \n" ) ;
7     else
8     {
9         printf ("To _ arxeio _ anoixe \n" ) ;
10        fclose (fp) ;
11    }
12    return 0 ;
13 }
```

---

### 1.1 Ανάγνωση κειμένου

Για την ανάγνωση απλού κειμένου από αρχείο υπάρχουν δύο τρόποι: ο πρώτος είναι να γίνει ανάγνωση του αρχείου γράμμα προς γράμμα. Ο δεύτερος είναι να γίνει ανάγνωση διαβάζοντας την μια γραμμή μετά την άλλη. Ο πρώτος τρόπος παρουσιάζεται στον αλγόριθμο 2, όπου γίνεται ανάγνωση όλων των γραμμών από ένα αρχείο και εμφανίζεται ο συνολικός αριθμός γραμμών και ο συνολικός αριθμός γραμμών. Για την ανάγνωση κάθε γράμματος χρησιμοποιήθηκε η συνάρτηση `fgetc()`. Αν θέλουμε να διαβάσουμε το αρχείο γραμμή προς γραμμή, τότε πρέπει να χρησιμοποιηθεί η συνάρτηση `fgets()`, όπως παρουσιάζεται στον αλγόριθμο 3. Η συνάρτηση `fgets()` παίρνει σαν ορίσματα το αλφαριθμητικό που θα διαβάσει, το μέγιστο πλήθος γραμμών που θα διαβάσει και τον δείκτη προς το αρχείο. Επιστρέφει 0 (NULL) αν τελειώσουν τα δεδομένα. Ωστόσο κατά την εκτέλεση του αλγορίθμου οι αλλαγές γραμμής εμφανίζονται δύο φορές. Αυτό γίνεται γιατί στο αλφαριθμητικό αντιγράφεται και ο χαρακτήρας αλλαγής γραμμής σαν τελευταίο γράμμα. Αυτή η παρενέργεια μπορεί να επιλυθεί διαγράφοντας το τελευταίο γράμμα από το αλφαριθμητικό, όπως παρουσιάζεται και στον αλγόριθμο 4.

### 1.2 Μορφοποιημένη είσοδος

Στην περίπτωση που θέλουμε να διαβάσουμε δεδομένα που δεν είναι απλώς γράμματα, όπως πχ αριθμούς, τότε απαιτείται η χρήση της συνάρτησης `fscanf`. Στο παράδειγμα του αλγορίθμου 5 στο αρχείο `numbers.txt` υπάρχουν σε κάθε γραμμή τρεις αριθμοί: ένας ακέραιος που είναι ο αριθμός μητρώου ενός σπουδαστή και δύο δεκαδικοί που είναι η βαθμολογία του σπουδαστή σε διάλεξη και εργαστήριο αντίστοιχα. Το πρόγραμμα εμφανίζει τον αριθμό μητρώου του σπουδαστή με την μεγαλύτερη μέση βαθμολογία. Το πλήθος των μαθητών είναι ίσο με 5. Το ίδιο πρόγραμμα μπορεί να γραφεί και με χρήση πίνακα δομών όπως παρουσιάζεται στον αλγόριθμο 6.

---

**Algorithm 2** Ανάγνωση γραμμάτων από αρχείο και εμφάνιση αριθμού γραμμάτων και αριθμού γραμμών.

---

```
1 # include <stdio.h>
2 int main()
3 {
4     FILE *fp=fopen("test.txt","r");
5     char ch;
6     int charcount=0;
7     int linecount=0;
8     if(fp==NULL)
9         printf("Lathos_sto_anoigma_tou_arxeiou\n");
10    else
11    {
12        printf("To_arxeio_anoixe\n");
13        do
14        {
15            ch=fgetc(fp);
16            if(ch!=EOF)
17            {
18                cout<<ch;
19                charcount++;
20                if(ch=='\n') linecount++;
21            }
22        }while(ch!=EOF);
23        printf("Synolika_grammata_%d\n",charcount);
24        printf("Synolikes_grammes_%d\n",linecount);
25        fclose(fp);
26    }
27    return 0;
28 }
```

---

---

**Algorithm 3** Εισαγωγή δεδομένων από αρχείο κειμένου με την χρήση της `fgets()`.

---

```
1 # include <stdio.h>
2 int main()
3 {
4     FILE *fp=fopen("test.txt","r");
5     char line[256];
6     int linecount=0;
7     if(fp==NULL)
8         printf("Lathos sto anoigma tou arxeiou\n");
9     else
10    {
11        printf("To arxeio anoixe\n");
12        while(fgets(line,255,fp))
13        {
14            cout<<"New_line_is_"<<line<<endl;
15            linecount++;
16        }
17        printf("Synolikes grammes_%d\n",linecount);
18        fclose(fp);
19    }
20    return 0;
21 }
```

---

---

**Algorithm 4** Ανάγνωση δεδομένων από αρχείο με την `fgets` και διαγραφή του χαρακτήρα αλλαγής γραμμής.

---

```
1 # include <stdio.h>
2 # include <string.h>
3 # include <iostream>
4 using namespace std;
5 int main()
6 {
7     FILE *fp=fopen("test.txt","r");
8     char line[256];
9     int linecount=0;
10    if(fp==NULL)
11        printf("Lathos sto anoigma tou arxeiou\n");
12    else
13    {
14        printf("To arxeio anoixe\n");
15        while(fgets(line,255,fp))
16        {
17            line[strlen(line)-1]='\0';
18            cout<<"New_line_is_"<<line<<endl;
19            linecount++;
20        }
21        cout<<"Synolikes_grammes_"<<linecount<<endl;
22        fclose(fp);
23    }
24    return 0;
25 }
```

---

---

**Algorithm 5** Είσοδος μαθητών από αρχείο και εύρεση της βαθμολογίας του καλύτερου σπουδαστή.

---

```
1 # include <stdio.h>
2
3 int main()
4 {
5     int code,maxcode,i;
6     double lab,lecture;
7     double average,maxaverage;
8     FILE *fp=fopen("numbers.txt","r");
9     for(i=0;i<5;i++)
10    {
11        fscanf(fp,"%d_%lf_%lf",&code,&lecture,&lab);
12        average=(lecture+lab)/2.0;
13        if(i==0 || average>maxaverage)
14        {
15            maxcode=code;
16            maxaverage=average;
17        }
18    }
19    fclose(fp);
20    printf("O_kalyteros_spoudastis_einai_me_kodiko:_%d\n",maxcode);
21    printf("O_kalyteros_bathmos_einai:_%lf\n",maxaverage);
22    return 0;
23 }
```

---

---

**Algorithm 6** Ανάγνωση πίνακα δομών από αρχείο.

---

```
1 # include <stdio.h>
2
3 typedef struct
4 {
5     int code;
6     double lab;
7     double lecture;
8 }Student;
9
10 double computeAverage(Student st)
11 {
12     return (st.lab+st.lecture)/2.0;
13 }
14
15 int main()
16 {
17     Student myclass[5];
18     Student beststudent;
19     double average;
20     int i;
21     FILE *fp=fopen("numbers.txt","r");
22     for(i=0;i<5;i++)
23     {
24         fscanf(fp,"%d_%lf_%lf",&myclass[i].code,
25             &myclass[i].lecture,&myclass[i].lab);
26         average=computeAverage(myclass[i]);
27         if(i==0 || average>computeAverage(beststudent))
28         {
29             beststudent=myclass[i];
30         }
31     }
32     fclose(fp);
33     printf("O_kalyteros_spoudastis_einai_me_kodiko:_%d\n",
34         beststudent.code);
35     printf("O_kalyteros_bathmos_einai:_%lf\n",
36         computeAverage(beststudent));
37     return 0;
38 }
```

---

---

**Algorithm 7** Εισαγωγή ζυγών αριθμών σε αρχείο.

---

```
1 # include <stdio.h>
2 int main()
3 {
4     FILE *fp;
5     int x;
6     int count=0;
7     fp=fopen("file.txt","w");
8     do
9     {
10         cout<<"Doste_zygo_arithmo_";
11         cin>>x;
12         if(x % 2 ==0 && x>0)
13         {
14             fprintf(fp,"%d\n",x);
15             count++;
16         }
17     }while(x>0);
18     fclose(fp);
19     printf("Synolikoi_zygoi_%d\n",count);
20     return 0;
21 }
```

---

### 1.3 Δημιουργία αρχείων

Μέσα από την γλώσσα μπορούμε να δημιουργήσουμε και αρχεία κειμένου, ώστε να αποθηκευτούν σε αυτά πληροφορίες. Στο παράδειγμα 7 ο χρήστης εισάγει ακέραιους αριθμούς μέχρι να εισάγει αρνητική τιμή. Κάθε ζυγός αριθμός αποθηκεύεται στο αρχείο file.txt

### 1.4 Αντιγραφή αρχείων

Με την γλώσσα C μπορούμε να αντιγράψουμε αρχεία, όπως στο παράδειγμα 8 όπου τα δεδομένα από το πρώτο αρχείο αντιγράφονται στο δεύτερο γράμμα προς γράμμα.



---

**Algorithm 8** Πρόγραμμα αντιγραφής αρχείων.

---

```
1 # include <stdio.h>
2 int main()
3 {
4     char sourcefile[100];
5     char destfile[100];
6     FILE *fin ,*fout ;
7     char ch;
8     printf("Doste_to_arxeio_pigis\n");
9     gets(sourcefile);
10    printf("Doste_to_arxeio_proorismou\n");
11    gets(destfile);
12    fin=fopen(sourcefile,"r");
13    if(!fin)
14    {
15        printf("To_arxeio_%s_den_yparxei\n",sourcefile);
16    }
17    else
18    {
19        fout=fopen(destfile,"w");
20        do
21        {
22            ch=fgetc(fin);
23            if(ch!=EOF)
24                fputc(ch,fout);
25        }while(ch!=EOF);
26        fclose(fin);
27        fclose(fout);
28    }
29    return 0;
30 }
```

---