



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΜΑΤΙΚΗΣ

# Μυριάνθης Αποστολίδης

2<sup>η</sup> Εργασία στο μάθημα **Λειτουργικά Συστήματα**

Ταύρος, XX Ιανουαρίου 2021

Περιεχόμενα	
<b>Άσκηση 2</b>	<b>2</b>
Κώδικας	2
Τρόπος Εκτέλεσης	3
Ενδεικτικές εκτελέσεις (screenshots):	3
Βασική εκτέλεση του προγράμματος	3
Παρατηρήσεις/σχόλια	3
Δημιουργία νέων διεργασιών και φυσιολογικός τερματισμός τους	3
Παρατηρήσεις/σχόλια	4
Δημιουργία νημάτων και φυσιολογικός τερματισμός τους	4
Παρατηρήσεις/σχόλια	5
Δημιουργία νημάτων και φυσιολογικός τερματισμός τους	5
Παρατηρήσεις/σχόλια	6
Γενικά Σχόλια/Παρατηρήσεις	6
Με δυσκόλεψε / δεν υλοποίησα	6
<b>Συνοπτικός Πίνακας</b>	<b>7</b>

## Άσκηση 2

### Κώδικας

Ο κώδικας της 2ης εργασίας που δημιουργήθηκε μαζί με τα σχόλια είναι:

```
#include <stdio.h>
#include <signal.h>
#include <stdlib.h>
#include <unistd.h>
#include <errno.h>
#include <string.h>
#include <fcntl.h>
```

```

#include <sys/types.h>
#include <dirent.h>
#include <sys/wait.h>
#include <pthread.h>

int words = 0;

//Function to count words

void *WordCounter(void *var)
{
    FILE *file;
    file = fopen(var, "r");
    char a;

    while ((a = fgetc(file)) != EOF)
    {

        /* Check words */
        if (a == ' ' || a == '\n' || a == '\0' || a == '\t')

            words++;
    }

    fclose(file);
    pthread_exit(0);
}

int main(int argc, char **argv)
{

    struct dirent *pDirent;
    DIR *pDir;
    char *filenames[100];
    int i;
    char ch;

    if (argc != 2)
    {
        pDir = opendir(".");
    }
    if (argc == 2){

```

```

        pDir = opendir(argv[1]);
    }

    // Ensure we can open directory.
    if (pDir == NULL)
    {
        printf("Cannot open directory '%s'\n", argv[1]);
        return 1;
    }

    // Process each entry.

    i = 0;
    while ((pDirent = readdir(pDir)) != NULL)
    {
        //Save the names of the files of the dir in an array
        filenames[i] = (pDirent->d_name);
        FILE *file;
        int pid, status;
        char path[1000];
        char c1[2] = "/";

        //Create the path to open the file
        strcpy(path, argv[1]);
        strcat(path, c1);
        strcat(path, filenames[i]);

        //Open file
        file = fopen(path, "r");

        char c;
        int bytes;
        //Read file char by char
        int counter = 0;
        while ((c = fgetc(file)) != EOF)
        {
            //Check if it is ascii
            if (c < 0 || c > 127)
            {
                counter = 1;
                break;
            }
        }
    }

```

```

}

if (counter == 1)
{
    //printf("File is not ascii and the path is %s\n",path);
}

else
{
    //printf("File is ascii and the path is %s\n",path);

    //Call fork
    if ((pid = fork()) == -1) /*check for error*/
    {
        perror("fork");
        exit(1);
    }

    if (pid == 0) /*The child process*/
    {
        //Create thread and use WordCounter function to find the words
        pthread_t tid;
        pthread_create(&tid, NULL, WordCounter, path);

        pthread_join(tid, NULL);

        FILE *fd;
        fd = fopen("output.txt", "a+");
        if (fd == NULL)
        {
            printf("Error opening file\n");
            exit(1);
        }

        //Check if file is empty
        if (words == 0)
        {
            return 0;
        }

        //Print to the file
        else
        {
            if (i == 0)

```

```

        {
            fprintf(fd, "PID      FILENAME      WORDS\n");
        }
        fprintf(fd, "%d      %s      %d\n", getpid(), pDirent->d_name, words);
    }
    fclose(fd);
    exit(0);
}
wait(NULL);
i++;
}
}

// Close directory and exit.
closedir(pDir);
}

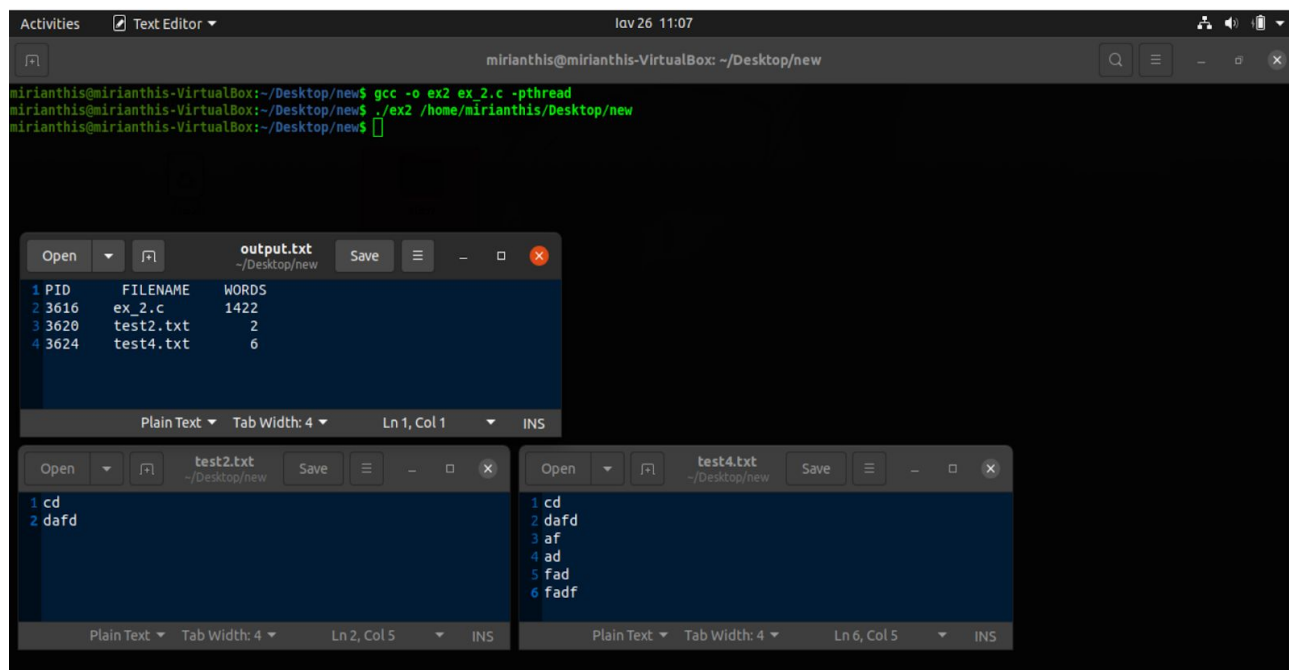
```

## Τρόπος Εκτέλεσης

Μέσα στον κατάλογο έχω 4 test files για να ελέγξω αν είναι σωστές ή λέςξεις.

## Ενδεικτικές εκτελέσεις (screenshots):

### Βασική εκτέλεση του προγράμματος



The screenshot shows a terminal window and three text editor windows. The terminal window displays the following commands and output:

```
mirianthis@mirianthis-VirtualBox: ~/Desktop/new$ gcc -o ex2 ex_2.c -pthread
mirianthis@mirianthis-VirtualBox:~/Desktop/new$ ./ex2 /home/mirianthis/Desktop/new
mirianthis@mirianthis-VirtualBox:~/Desktop/new$
```

The first text editor window, titled "output.txt", shows the following output:

PID	FILENAME	WORDS
3616	ex_2.c	1422
3620	test2.txt	2
3624	test4.txt	6

The second text editor window, titled "test2.txt", shows the following content:

```
1 cd
2 dafd
```

The third text editor window, titled "test4.txt", shows the following content:

```
1 cd
2 dafd
3 af
4 ad
5 fad
6 fadf
```

### Παρατηρήσεις/σχόλια

Για να κάνεις compile το αρχείο στο gcc πρέπει να βάλεις και το -pthread αλλιώς δεν αναγνωρίζει τα threads.

❏ Δημιουργία νέων διεργασιών και φυσιολογικός τερματισμός τους

```
mirianthis@mirianthis-VirtualBox:~/Desktop/new$ gcc -o ex2 ex_2.c -pthread
mirianthis@mirianthis-VirtualBox:~/Desktop/new$ ./ex2 /home/mirianthis/Desktop/new
Creating new child procces pid:3804
Child ended normally pid:3804
Creating new child procces pid:3806
Creating new child procces pid:3808
Child ended normally pid:3808
Creating new child procces pid:3810
Creating new child procces pid:3812
Child ended normally pid:3812
Creating new child procces pid:3814
Child ended normally pid:3814
mirianthis@mirianthis-VirtualBox:~/Desktop/new$
```

Παρατηρήσεις/σχόλια

Γενικά Σχόλια/Παρατηρήσεις

Καλή άσκηση χρειάζεται αρκετό ψάξιμο για να υλοποιηθεί

Με δυσκόλεψε / δεν υλοποίησα

Δυσκολεύτηκα να υλοποιήσω τα threads



# Συνοπτικός Πίνακας

2η Εργασία		
Λειτουργία	Υλοποιήθηκε (ΝΑΙ/ΟΧΙ/ΜΕ ΡΙΚΩΣ)	Συνοπτικές Παρατηρήσεις
Βασική διεργασία η οποία ελέγχει τις παραμέτρους και βρίσκει τα ascii αρχεία	ΝΑΙ	
Δημιουργία νέων διεργασιών και φυσιολογικός τερματισμός τους	ΝΑΙ	
Δημιουργία νημάτων και φυσιολογικός τερματισμός τους	ΝΑΙ	
Ομαλή εκτέλεση προγράμματος, error handling, τεκμηρίωση	ΝΑΙ	