

## תרגיל 3 C בסביבת יוניקס

שאלה 1)

כתבו תכנית המקבלת כארגומט (לא ע"י cin... אלא ע"י argv & argc) מהמשתמש שמות של שני אנשים, שם פרטי ושם משפחה, והגילאים שלהם בודקת האם נכנסו 6 פרמטרים כנדרש. מבקשת ומקבלת מהמשתמש עוד שם של אדם (שם פרטי שם משפחה וגיל ע"י שימוש בפקודה gets) יוצרת 3 קבצים בראשון מכניסה את השם הראשון ואת גילו כנ"ל לקובץ השני והשלישי, מדפיסה למסך את הגילאים שלהם בלבד.

**useful function:** strtoc, open (not fopen), read, atoi

נניח כי שם קובץ ההרצה הוא xyz :

Propt>xyz aaa AAA 28 bbb BBB 32

File1 aaa AAA 28	File2 bbb BBB 32	File3 ccc CCC 36
---------------------	---------------------	---------------------

שאלה 2)

כתבו תכנית המקבלת שלושה פרמטרים של מספרים שלמים ומדפיסה למסך את הממוצע שלהם מספרים הנ"ל.

**useful function:** atoi

נניח לדוגמא כי שם קובץ ההרצה הוא abc :

Propt>abc 28 32 36

Propt>32

שאלה 3)

עליכם להריץ את התכנית שבשאלה 1 ודרך pipe להעביר את מה שהיא מדפיסה למסך כקלט לתכנית שבשאלה 2. כתבו שתי תוכניות שונות; בראשונה את ה-pipe עליכם לממש ע"י dup בשנייה עליכם לממש את ה-pipe ע"י dup2.

**useful function:** atoi, fork, dup, dup2, execv (not execl)

נניח לדוגמא כי שם קובץ ההרצה הוא pipe\_exec :

Propt> pipe\_exec

Propt>32

שאלה 4)

### Simple mini shell

Implement a mini shell. The mini shell should accept simple commands from the stdin and execute them using the combination of fork and exec. The shell should terminate when the user types exit. The minishell should print a prompt (e.g. minishell >). The minishell should receive as optional argument -debug to print some detail information about the execution of the father and the child processes.

```
gcc minishell.c -o minishell
```

```
./minishell -debug
```

```
INFO: Father started PID[810]
```

```
Welcome in my mini shell. Type "exit" to terminate.
```

```
minishell> pwd
```

```
INFO: Child started PID[811] command "pwd"
```

```
/home/fredj/work/unix/ex2
```

```
INFO: Child with PID[811] terminated, continue waiting commands
```

```
minishell> ll -al
```

INFO: Child started PID[812] command "ls -al"

ls: command not found

INFO: Child with PID[812] terminated, continue waiting commands

minishell> exit

INFO: Father will terminate

**Useful commands:** fgets, strtok, fork, exec, sprintf

שאלה 5)

### Extended mini shell

Extend the mini shell to support commands with | (pipe). To simplify, only consider commands with 1 pipe to redirect the stdout between two processes. Think carefully how to create the children processes to handle the pipe, and do not forget to close all the unused file descriptors

```
gcc minishell2.c -o minishell2
```

```
./minishell2 -debug
```

Welcome in my extended mini shell. Type "exit" to terminate.

```
minishell2> pwd
```

INFO: No pipe detected, creating child for command "pwd"

INFO: Child started PID[811] command "pwd"

```
/home/fredj/work/unix/ex2
```

INFO: Child with PID[811] terminated, continue waiting commands

```
minishell2> cat *.c | grep include
```

INFO: Pipe detected. Command 1: "cat \*.c" and Command 2: "grep include"

INFO: Making pipe

INFO: Child started PID[812] command "cat \*.c"

INFO: Child started PID[813] command "grep include"

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

```
#include <math.h>
```

INFO: Child with PID[812] terminated

INFO: Child with PID[813] terminated

```
minishell2>exit
```

INFO: Father will terminate

**Useful commands:** fgets, strtok, fork, exec, pipe, dup, dup2, open, close