

NOM		DATA	<b>27 / 05 / 2016</b>	QUALIFICACIÓ
ÀREA/MATÈRIA	<b>ASIX/DAW - M03 - UF3</b>	CURS	<b>2015 - 2016</b>	

**Escolliu preguntes de l'examen que sumin 10 punts (hi ha 14 punts repartits en 10 preguntes). Hi ha quatre blocs : 1-3, 4-5, 6-9 i 10. Feu captures de pantalla per a documentar l'examen.**

**1)(1 punt) Compileu *wifi.c* i executeu-lo. Cal tenir al mateix directori d'execució l'arxiu *wifiList.txt*.**

**2)(1 punt) Traieu els comentaris de les línies 37, 46 i 56. Comenteu amb *//* tots els *printf* actuals del codi. Quan es trobi la paraula *ESSID* afegiu un *printf* que mostri la cèl·lula, l'SSID i el nivell de potència del senyal wifi. Desenvolpeu *ex02.c* que extreu la informació de l'arxiu *wifiList.txt* generat i la presenta en aquest ordre i format:**

```
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex02
```

```
ESSID (cell 1) : IoT-eCat -38dBm
ESSID (cell 2) : _AUTO_ONOWiFi -79dBm
ESSID (cell 3) : _ONOWiFi -80dBm
ESSID (cell 4) : HP-Print-D8-Deskjet 2540 series -53dBm
ESSID (cell 5) : ONOF5AC -46dBm
ESSID (cell 6) : _AUTO_ONOWiFi -48dBm
ESSID (cell 7) : _ONOWiFi -48dBm
ESSID (cell 8) : MOVISTAR_9FCA -72dBm
ESSID (cell 9) : INT_U9T7H9 -72dBm
ESSID (cell 10) : WLAN_8F -81dBm
ESSID (cell 11) : Valledupar -73dBm
ESSID (cell 12) : TRAC -81dBm
ESSID (cell 13) : vodafone5FA2 -78dBm
```

```

47     }
48     p = strstr(szLinia,"ESSID:");
49     if(p){
50         p[strlen(szLinia,"\\n")-p] = '\\0';
51         //printf ("%s",p);
52         p = strstr(p,"\\n") + 2;
53         /*(strstr(p,"\\n")) = '\\0';
54         p[strlen(p)-1] = '\\0';
55         //printf("#s# ",p);
56         strcpy(sf.szESSID,p);
57         //printf("\\n");
58         printf("ESSID (cell %d) : %s %ddBm\\n",sf.nCell,sf.szESSID,sf.nPower);
59     }
60     }
61     fclose(f);
62 }else
63     printf("Sembla que wifiList.txt no està a aquest directori \\n");

```

**3)(1 punt) Desenvolpeu *ex03.c* que es comporta com *ex02.c* però si se li passa com argument el nom de la xarxa ens presenta el seu nivell de potència:**

```
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex03 WLAN_8F
-81dBm
```

**i en cas de no existir :**

```
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex03 unaAltraXarxa
unaAltraXarxa no detectada
```

```

56     strcpy(sf.szESSID,p);
57     //printf("\\n");
58     if(argc == 1)
59         printf("ESSID (cell "
60     if(argc == 2){
61         if(!strcmp(sf.szESSID
62             printf("%ddBm\\n"
63                 bDetectada = 1;
64         }
65     }
66     }
67     }
68     if(!bDetectada)
69         printf("%s no detectada\\n",a
70         fclose(f);
71 }else
72     printf("Sembla que wifiList.txt i

```

4)(2 punts) Implementeu *ex04.c* que llegeix la primera línia d'un arxiu passat com a argument. En cas de que l'arxiu es pugui llegir retornarà 0 al sistema. Si l'arxiu passat com a argument no existeix o no hi ha arguments el programa informarà i retornarà 1 al sistema. Funcionament esperat al terminal:

```
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex04
Cal l'argument de l'arxiu a llegir
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex04 ex04.perl
L'arxiu ex04.perl no existeix
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex04 ex04.py
#!/usr/bin/python
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex04 ex04.sh
#!/bin/bash
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$
```

5)(1 punt) Implementeu *ex05.c* que llegeix la darrera línia d'un arxiu passat com a argument. En cas de que l'arxiu es pugui llegir retornarà 0 al sistema. Si l'arxiu passat com a argument no existeix o no hi ha arguments el programa informarà i retornarà 1 al sistema. Funcionament esperat al terminal:

```
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex05 ex04.py
i+=1
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex05 ex04.sh
# the scope of this training unit.
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex05 ex04.perl
L'arxiu ex04.perl no existeix
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex05
Cal l'argument de l'arxiu a llegir
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$
```

6)(1 punt) Implementeu *ex06.c* prenent coma a base l'arxiu *valls.c*, que llegeixi el nombre de valors reals (*float*) a introduir i després els presenti amb quatre decimals. Feu les modificacions oportunes a les tres funcions. Funcionament esperat al terminal:

```
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex06
Quantes dades a entrar a fV?: 2
**** vIntrodueixValors ****
p[0]?: -1.34567
p[1]?: 3.14159
**** vVisualitzaNumeros ****
p[0]: -1.3457
p[1]: 3.1416
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$
```

7)(1 punt) Implementeu *ex07.c* prenent coma a base l'arxiu *ex06.c*, que faci servir la funció *calloc()* en comptes de *malloc()*. El funcionament esperat al terminal és exactament el mateix que el d'*ex05.c*

8)(1 punt) Implementeu *ex08.c* prenent coma a base l'arxiu *ex07.c*, que enregistri a l'arxiu *numeros.reals* els números entrats en format real (*float*). Caldrà que implementeu la funció *int bEnregistraNumerosReals( float \*p, int nQ , char\* szNomArxiu)* i la podeu cridar així *bEnregistraNumerosReals( fV , nQuants , NOM\_ARXIU)*; Funcionament esperat al terminal:

```
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex08
Quantes dades a entrar a fV?: 3
**** vIntrodueixValors ****
p[0]?: 1.23456
p[1]?: 9.87
p[2]?: 3.141592653589793
**** vVisualitzaNumeros ****
p[0]: 1.2346
p[1]: 9.8700
p[2]: 3.1416
**** bEnregistraNumerosReals ****
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ cat numeros.reals
1.234560
9.870000
3.141593
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$
```

9)(2 punts) Implementeu `ex09.c` que llegeixi l'arxiu `numeros.reals` en format real (*float*). Caldrà que implementeu la funció `int nPresentaNumerosReals(char* szNomArxiu)` que retorna el nombre de números reals llegits. Heu d'incloure l'arxiu de capçalera `stdlib.h`. Funcionament esperat al terminal:

```
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$ ./ex09
1.234560
9.870000
3.141593
Hi havia 3 números reals
ecat@debian8:~/Documents/m3/uf3/20150527ex/sol$
```

Recordeu que per passar cadenes de caràcters a reals heu de fer servir la funció `atof`

Exemple de funció principal:

```
int main(){
    int nQuants;

    if(!(nQuants = nPresentaNumerosReals(NOM_ARXIU))){
        printf("L'arxiu %s no existeix\n",NOM_ARXIU);
        return 1;
    }
    printf("Hi havia %d números reals\n",nQuants);
    return 0;
}
```

10)(3 punts) L'horari escolar d'un grup de cicles formatius és de 6 hores diàries de dilluns a divendres. Feu `ex09.c` com a aplicació que ens permeti introduir les dades d'aquest horari.

Per cada hora haureu de definir:

- nom de l'assignatura.
- nom del professor.
- número d'aula.

L'horari s'emmagatzema en una estructura de dades anomenada `horari`. Trieu l'estructura de dades més adient per emmagatzemar aquesta informació i que tingui les següents funcionalitats:

- a) `vEmplenaHorari()`. Ha de permetre introduir totes les hores.
- b) `vEmplenaHora()`. Permet introduir les dades d'una determinada hora d'un dia.
- c) `vLlistarHorari()`. Treu per pantalla l'horari.
- d) `vModificarHora()`. Permet canviar les dades d'una determinada hora d'un dia.

***Molta sort a totes i tots !!!!***