

Programmation en C

Corrigé TD10

```
1 /* Table ASCII
2 ----- */
3
4 #include <stdio.h>
5
6 int main() {
7     char c;
8
9     for(c=0; c<128; c++)
10         printf("%d --> %c \n",c,c);
11     return 0;
12 }
```

```
1 /* Renversement d'un mot
2 ----- */
3
4 #include <stdio.h>
5 #include <stdlib.h>
6 #include <string.h>
7
8 char *reverse(char *S) {
9     int i;
10    int l = strlen(S);
11    char *R = malloc((l+1)*sizeof(char));
12
13    for (i=0;i<l;i++)
14        R[i] = S[l-1-i];
15    R[l] = '\0';
16    return R;
```

```

17 }
18
19 int main(int argc,char *argv[]) {
20     char *S;
21
22     /* verification presence du parametre */
23     if(argc != 2) {
24         printf("Entrer un mot\n");
25         exit(0);
26     }
27
28     S = reverse(argv[1]);
29     printf("%s --> %s \n",argv[1],S);
30
31     return 0;
32 }

1 /* Test palindrome ?
2 ----- */
3
4 #include <stdio.h>
5 #include <stdlib.h>
6 #include <string.h>
7
8 int palindrome(char *S) {
9     int i;
10    int l = strlen(S);
11    int t = 1;
12
13    for (i=0;i<l;i++)
14        if (S[i] != S[l-1-i]) t=0;
15
16    return t;
17 }
18
19 int main(int argc,char *argv[]) {
20
21     /* verification presence du parametre */

```

```

22  if(argc != 2) {
23      printf("Entrer un mot\n");
24      exit(0);
25  }
26
27  if (palindrome(argv[1]))
28      printf("%s est un palindrome \n",argv[1]);
29  else
30      printf("%s n'est pas un palindrome \n",argv[1]);
31
32  return 0;
33  }

1  /* Decryptage d'une date de naissance
2  ----- */
3
4  #include <stdio.h>
5  #include <stdlib.h>
6  #include <string.h>
7  #define ANNEE 2009
8
9  void decodeDate(char *date) {
10     int x,jour,mois,annee;
11     char ch[5];
12
13     /* verification de la longueur de la chaine */
14     x = strlen(date);
15     if (x != 10) {
16         printf("une date AAAA-MM-JJ comporte 10 caracteres\n");
17         exit(0);
18     }
19
20     /* decodage */
21     strncpy(ch,date,4);ch[4]='\0';
22     annee = atoi(ch);
23     strncpy(ch,date+5,2);ch[2]='\0';
24     mois = atoi(ch);
25     strncpy(ch,date+8,2);ch[2]='\0';

```

```

26  jour = atoi(ch);
27  printf("Jour du mois : %d\n",jour);
28  printf("Mois : %d\n",mois);
29  printf("Annee : %d\n",annee);
30  printf("Age : %d\n",ANNEE-annee);
31 }
32
33 int main(int argc,char *argv[]) {
34  /* verification presence du parametre */
35  if (argc != 2) {
36  printf("Donner une date sous la forme AAAA-MM-JJ\n");
37  exit(0);
38  }
39
40  decodeDate(argv[1]);
41
42  return 0;
43 }

1  /* Minuscules -> Majuscules
2  ----- */
3
4  #include <stdio.h>
5  #include <stdlib.h>
6
7  void upper(FILE *in,FILE *out) {
8  char c,d;
9
10 while (feof(in)==0) {
11  c = fgetc(in);
12  if (c!=EOF) {
13  d = c;
14  if ((c > 96) && (c < 123)) d = c - 32;
15  fputc(d,out);
16  }
17 }
18 }
19

```

```

20 int main(int argc,char *argv[]) {
21     FILE *in,*out;
22
23     if (argc != 3) {
24         printf("donner deux noms de fichiers\n");
25         exit(0);
26     }
27
28     in = fopen(argv[1],"r");
29     if (in == NULL) {
30         printf("impossible d'ouvrir %s en lecture\n",argv[1]);
31         exit(0);
32     }
33     out = fopen(argv[2],"w");
34     if (out == NULL) {
35         printf("impossible d'ouvrir %s en ecriture\n",argv[2]);
36         exit(0);
37     }
38     upper(in,out);
39     fclose(in);fclose(out);
40     return 0;
41 }

1 /* Commande unix "wc"
2  ----- */
3
4 #include <stdio.h>
5 #include <stdlib.h>
6
7 /* Le resultat est identique a la commande wc d'Unix */
8 /* sur les fichiers textes, mais pas forcement sur */
9 /* des fichiers contenant des caracteres speciaux. */
10
11 int *wc(FILE *f) {
12     int nl,nm,nc;
13     char c,cp;
14     int *n;
15

```

```

16  /* init nb de lignes, de mots et de caracteres */
17  nl = nm = nc = 0;
18  /* init caractere precedent */
19  cp=' ';
20
21  while (feof(f)==0) {
22      /* lecture caractere */
23      c= fgetc(f);
24
25      if (c!=EOF) {
26          /* incrementation nb de caractere */
27          nc ++;
28
29          /* detection caractere suivant
30             un espace sans etre un \n ou un \t */
31          if ((c!=' ') && (cp==' '))
32              && (c!='\n') && (c!='\t')) nm++;
33
34          /* detection caractere suivant
35             une tabulation sans etre un \n ou un \t */
36          if ((c!=' ') && (cp=='\t'))
37              && (c!='\n') && (c!='\t')) nm++;
38
39          /* detection caractere suivant
40             un \n sans etre un \n ou un \t */
41          if ((c!=' ') && (cp=='\n'))
42              && (c!='\n') && (c!='\t')) nm++;
43
44          /* detection retour a la ligne */
45          if (c=='\n') nl++;
46
47          /* sauvegarde caractere precedent */
48          cp = c;
49      }
50  }
51  n = malloc(3 * sizeof(int));
52  n[0] = nl; n[1] = nm; n[2] = nc;
53  return n;

```

```

54 }
55
56 int main(int argc,char *argv[])
57 {
58     int *n;
59     FILE *f;
60
61     system("date");
62
63     if (argc != 2) {
64         printf("donner un nom de fichier\n");
65         exit(0);
66     }
67     f = fopen(argv[1],"r");
68     if (f == NULL) {
69         printf("impossible d'ouvrir %s\n",argv[1]);
70         exit(0);
71     }
72     n = wc(f);
73     fclose(f);
74     printf("%7d %7d %7d %s\n",n[0],n[1],n[2],argv[1]);
75     return 0;
76 }

1 /* Decryptage d'une date de naissance
2 ----- */
3
4 #include <stdio.h>
5 #include <stdlib.h>
6 #include <string.h>
7 #include <time.h>
8
9 void decodeDate(char *date) {
10     int x,jour,mois,annee;
11     char ch[5];
12     time_t rawtime;
13     struct tm *timeinfo;
14     int ANNEE;

```

```

15
16  /* verification de la longueur de la chaine */
17  x = strlen(date);
18  if (x != 10) {
19      printf("une date AAAA-MM-JJ comporte 10 caracteres\n");
20      exit(0);
21  }
22
23  time(&rawtime);
24  timeinfo=localtime(&rawtime);
25  ANNEE=1900+timeinfo->tm_year;
26  printf("L'annee est %d\n",ANNEE);
27
28  /* decodage */
29  strncpy(ch,date,4);ch[4]='\0'; annee = atoi(ch);
30  strncpy(ch,date+5,2);ch[2]='\0'; mois = atoi(ch);
31  strncpy(ch,date+8,2);ch[2]='\0'; jour = atoi(ch);
32  printf("Jour du mois : %d\n",jour);
33  printf("Mois : %d\n",mois);
34  printf("Annee : %d\n",annee);
35  int age=ANNEE-annee;
36  printf("Age : %d\n",ANNEE-annee);
37 }
38
39 int main(int argc,char *argv[]) {
40     /* verification presence du parametre */
41     if (argc != 2) {
42         printf("Donner une date sous la forme AAAA-MM-JJ\n");
43         exit(0);
44     }
45     decodeDate(argv[1]);
46     return 0;
47 }

```