

TP n°7 : Threads et parallélisme

SOLUTIONS (ne pas distribuer aux étudiants)

Exercice 1 – Les bases des threads

Solution : Voir corrigé du TD.

Exercice 2 – Pour aller un peu plus loin...

Solution :

```
1)
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <string.h>

//Taille maximale d'un mot en francais ("anticonstitutionnellement")
#define MAX_THREADS 25

char mot[26];

// Code des threads
void* code_thread_lettre(void *lettre)
{
    printf("%c est de rang %d\n", (char)lettre, (char)lettre-'a'+1);
    return 0;
}

// Code du main
int main()
{
    int ret;
    pthread_t thread_lettre[MAX_THREADS];
    int i;

    //lecture du mot
    printf("Entrez un mot : ");
    scanf("%s",mot);
    printf("le mot est \"%s\" possède %d carac \n",mot,strlen(mot));

    // création d'un thread par lettre du mot
    for (i=0;i<strlen(mot);i++)
    {
        ret = pthread_create(&thread_lettre[i],NULL, code_thread_lettre, mot[i]);
        if (ret !=0)
        {
            fprintf(stderr, "Erreur lors de la création du thread n°%d\n",i+1);
            exit(1);
        }
    }
}
```

```
// Attente de la terminaison des threads
i=0;
while(i<strlen(mot))
{
    printf("Attente de la terminaison du thread n°%d...\n", i+1);
    pthread_join(thread_lettre[i],NULL);
    printf("Thread n°%d terminé...\n", i+1);
    i++;
}
printf("Fin des threads...\n");
return 0;
}
```