**TD4 :**

1) Premier appel à la fonction

Une image contenant texte

Description générée automatiquement

En enlevant pthread\_join



La valeur renvoyée est différente et les print écrits dans la fonction fn\_thread n’apparaissent pas.

Un thread ne finit que si un pthread\_exit est appelé ou si le programme s’arrête.



Avec le exit, on quitte le programme après le print.

Si on met le exit avec le pthread\_join, le programme finit dès qu’on y arrive.

Un exit termine le programme alors qu’un pthread\_exit finit seulement le thread.

2) 3) Code

4) On ne peut pas créer autant de processus qu’on veut. Une image contenant texte

Description générée automatiquement

5) Il est plus rapide de créer des threads que des processus.

Une image contenant texte

Description générée automatiquement

6) Commandes

7) En passant l'adresse on récupère une valeur qui peut être modifiée dans le main, on corrige ça en ne passant que la valeur de i.

8) Par moment il reste -1 allumettes. La grande consommation de ressources s’expliquent par le fait que 2 processus tournent en même temps et font des modifications sur les mêmes choses.

Sans le sleep, à plusieurs reprises le joueur prend 0 allumettes à la suite parce que le jeu n’est pas réinitialisé.

11) Processus : sur Unix, l’utilisation semble plus efficace que sur windows.

Threads : sur Unix l’utilisation me parait plus simple d’utilisation mais sur Windows j’ai l’impression d’avoir plus de liberté