

À vos marques... Partez !

Écrivez un programme qui génère n threads, puis attend une saisie au clavier (par exemple avec `getchar`) pour débloquent les n threads qui attendent avec une barrière. Dès lors, chacun d'eux affiche son numéro i ($0 \leq i < n$). Un seul des threads (mais pas forcément toujours le même, ce ne doit pas être déterministe) doit afficher la phrase « je suis élu ». Enfin, tous les threads se terminent et le thread principal peut ensuite afficher la phrase « tous les threads ont terminé ».

Par exemple :

```
> ./partez 4                                # 4 threads sont générés
Saisie au clavier :                          # attente ⇒ il suffit d'appuyer sur [Entrée]
Thread 3 parti
Thread 1 parti, je suis élu                  # félicitations !
Thread 0 parti
Thread 2 parti
Tous les threads ont termine                # bien sûr, il faut s'assurer qu'ils ont tous bien terminé...
```

Vous n'utiliserez bien sûr aucune variable globale.

Bonus : peut-il arriver que le signal du départ soit donné alors que tous les threads n'ont pas encore démarré ?