

Partie PF :

→ Choix de l'implémentation de l'état :

- Pourquoi une liste d'entiers ? Trop compliqué d'après eux
- Module ? Pas la bonne solution
- Pourquoi pas un tableau ? Il faut se méfier mais c'est possible, faire attention aux références qui sont modifiées dans des programmes différents...
- Le plus simple : un arbre binaire de recherche comme vu en TP.

→ Pourquoi il y a un warning dans `faire_un_pas` ?

- On fait un `match |0 |1` et Ocaml pense qu'on ne couvre pas tous les cas. Vu que notre type est bien implémenté il n'y a pas à s'inquiéter on peut mettre `match |0 |_` pour « couvrir tous les cas »

→ Il m'a demandé d'expliquer nos fonction `read` et `modify` qui parcourent notre liste d'entiers pour représenter l'état ;

→ Dans notre fonction `affec`, pourquoi on fait un `read` dans la dernière ligne ?

→ Que fait notre fonction `terminal c` ?

→ Pourquoi dans la fonction `exec` on a un `| Inter ...` ?

Partie LT :

→ Dans `sos_pcarre_n`, pourquoi on fait un `SOS_if_false` ? Comment on aurait pu évaluer autrement ?

→ Dans cette même ligne pourquoi il y a un `eapply` et pas tout simplement un `apply` ?

- C'est juste une erreur car le `apply` aurait suffi

→ Expliquer `eqnatb_refl`. Que fait-elle en détail ?