TME numéro 3

(intermède musical)

Exercice 1 Reprenons le fichier `aide_client.mli`. Si vous le lisez vous verrez qu'il y a principalement deux types de fonctions décrites dedans : celles dont le préfixe est « `tme2_` » et celles commençant par « `aide_` ». Ces fonctions réalisent exactement les mêmes actions mais elles n'ont pas les mêmes arguments en entrée :

TME 2	
nom de fonction	type
`tme2_get_balles_a_afficher`	`unit -> Jeu.eltJeu list`
`tme2_get_vaisseaux_a_afficher`	`unit -> Jeu.eltJeu list`
`tme2_get_plateau_a_afficher`	`unit -> Jeu.eltJeu list`
`tme2_get_obstacles_a_afficher`	`unit -> Jeu.eltJeu list`
`tme2_recupere_action_joueur`	`unit -> unit`
`tme2_connexion_serveur`	`string -> string -> Graphics.color`

TME 3	
nom de fonction	type
`aide_get_balles_a_afficher`	`in_channel -> out_channel -> Jeu.eltJeu list`
`aide_get_vaisseaux_a_afficher`	`in_channel -> out_channel -> Jeu.eltJeu list`
`aide_get_plateau_a_afficher`	`in_channel -> out_channel -> Jeu.eltJeu list`
`aide_get_obstacles_a_afficher`	`in_channel -> out_channel -> Jeu.eltJeu list`
`aide_recupere_action_joueur`	`in_channel -> out_channel -> unit`
`aide_connexion_serveur`	`string -> string -> Graphics.color * in_channel * out_channel`

Les `in_channel` et `out_channel` sont respectivement le canal à partir duquel on lit les informations provenant du serveur et le canal par lequel on envoie des informations au serveur `ocamlpilotd`. C'est la fonction `aide_connexion_serveur` qui renvoie le `in_channel` et le `out_channel` utilisés par toutes les autres fonctions.

Modifiez le programme que vous aviez obtenu en fin de TME précédent de manière à utiliser ces nouvelles fonctions plutôt que les anciennes. Vous devez à nouveau obtenir une partie client de votre projet OcamlPilot. Pensez à mettre à jour vos commentaires OcamlDoc.

Exercice 2 Petite récréation : écrivez une fonction `test : string -> unit` qui prend en argument une chaîne de caractères prenant l'une des formes suivantes :

– nombre `xxx`
– string `yyy`
– couple `xxx zzz`

où `xxx` et `zzz` sont des `int` et `yyy` est une `string`, et qui affiche, selon les cas, ces `int` ou ces `string`. Ainsi l'appel `(test "nombre 34")` affichera 34 à l'écran et l'appel `(test "couple 12 56")` affichera 12 56, etc. Pensez à utiliser la fonction `sscanf` du module `Scanf`.