

Nom	
Prénom	

Note

S1# – Algorithmique
Contrôle n° 1 (C1)
3 mai 2021
Feuilles de réponses

I	
II	
III	
IV	
V	

Réponses 1 (Une pincée de listes... – 5 points)

1. Ecrire les opérations *tête*, *premier*, *fin* et *cons* :

2. Ecrire l'opération *insérer* :

Réponses 2 (Vérifier la présence – 4 points)

La fonction `check_occ x n list` vérifie si l'élément `x` est présent au moins `n` fois dans la liste `list`.

Fonction CAML :

Réponses 3 (Avant dernier – 4 points)

Spécifications :

La fonction `insert_penultimate x list` insère l'élément `x` en avant-dernière position de la liste `list`. La fonction déclenche une exception `Invalid_argument` si la liste donnée en paramètre est vide.

Fonction CAML :

Réponses 4 (find2 – 5 points)

1. Spécifications :

`find2 p [a1; a2; ···; an] [b1; b2; ···; bn]` retourne le premier couple d'éléments qui vérifie le prédicat p : la première paire d'éléments (a_i, b_i) tel que $p a_i b_i$ est vrai. Elle déclenche une exception `Failure` si aucun couple (a_i, b_i) tel que $p a_i b_i$ est vrai n'a été trouvé ou si les deux listes sont de longueurs différentes.

Fonction CAML :

2. Spécifications :

La fonction `first_shared 11 12` retourne le premier élément présent dans les deux listes à la même position.

Fonction CAML :

Réponses 5 (Mystery – 3 points)

1. Évaluation :

```
# let rec mystery a b c = match a with
  []      -> b
  | d::e -> let r = (mystery e b c) in c d r
;;
```

2. Évaluations :

```
# mystery [] 12 (let c x y = x+y in c) ;;
```

```
# mystery [1; 2; 3; 4; 5] 0 (function x -> function y -> x+y) ;;
```
