

Exercice 3 (Insertion à la $i^{\text{ème}}$ place – 5 points)

Écrire la fonction `insert_nth` x i `list` qui insère la valeur x à la $i^{\text{ème}}$ place dans la liste `list`. La fonction devra déclencher une exception `Invalid_argument` si i est négatif ou nul, ou une exception `Failure` si la liste est trop courte.

Exemples d'application :

```
# insert_nth 0 5 [1; 2; 3; 4; 5; 6; 7; 8; 9];;
- : int list = [1; 2; 3; 4; 0; 5; 6; 7; 8; 9]

# insert_nth 0 10 [1; 2; 3; 4; 5; 6; 7; 8; 9];;
- : int list = [1; 2; 3; 4; 5; 6; 7; 8; 9; 0]

# insert_nth 0 12 [1; 2; 3; 4; 5; 6; 7; 8; 9];;
Exception: Failure "out of bound".

# insert_nth 0 (-2) [1; 2; 3; 4; 5; 6; 7; 8; 9];;
Exception: Invalid_arg "negative rank".
```

[illegible]

Écrire la fonction `search_both list a b` qui vérifie si les deux valeurs a et b distinctes sont présentes dans la liste `list`.

```
# search_both [12; 5; -4; 0; 7; 21; 3] 5 0;;
- : bool = true
# search_both [12; 5; -4; 0; 7; 21; 3] 21 (-4);;
- : bool = true
# search_both [12; 5; -4; 0; 7; 21; 3] 0 42;;
- : bool = false
```

This image shows a full page of blank graph paper. The background is a very light gray color. Overlaid on this background is a grid of thin, medium-gray lines. The grid consists of small, identical squares that cover the entire area of the page. There are no margins, text, or other markings on the paper.

Exercice 5 (Mystery – 2 points)

Soit la fonction `go` définie ci-dessous

```
let go = function
  [] -> []
| e::list ->
  let rec what x = function
    [] -> []
  | e::list -> (e * x)::(what e list)
  in
  what e list;;
```

Donner les résultats des évaluations successives des phrases suivantes.

go [1; 1; 1; 1; 1] ;;

go [42] ;;

go [1; 2; 3; 4; 5] ;;

go [2; 21; 2; 21; 2; 21] ;;

