

(\* 課題3.1 \*)

(\* 1 \*)

```
let rec num_of (l, x) =
  match l with
  [] -> 0
  | h::rest ->
    if h = x then 1 + num_of (rest, x)
    else num_of (rest, x);;
num_of ([1;2;1;3;3;],3);;
```

(\* 実行結果 \*)

```
val num_of : 'a list * 'a -> int = <fun>
- : int = 2
```

(\* 2 \*)

```
let num_of (l, x) =
  let rec num (l, x, a) =
    match l with
    [] -> a
    | h::rest ->
      if h = x then num (rest, x, a+1)
      else num (rest, x, a) in
  num (l, x, 0);;
num_of ([1;2;1;3;3;],3);;
```

(\* 実行結果 \*)

```
val num_of : 'a list * 'a -> int = <fun>
- : int = 2
```

(\* 課題3.2 \*)

```
let sum_pair x =
  let rec sum (x, a, b) =
    match x with
    [] -> (a, b)
    | h::rest ->
      if h < 0 then sum (rest, a+h, b)
      else sum (rest, a, b+h) in
  sum (x, 0, 0);;
sum_pair [-2; 0; 3; -1; 2; 1];;
```

(\* 実行結果 \*)

```
val sum_pair : int list -> int * int = <fun>
- : int * int = (-3, 6)
```

(\* 課題3.3 \*)

```
let prefix_sum x =
  let rec prefix (x, a, arr) =
    match x with
    [] -> arr
    | h::rest -> prefix (rest, a + h, arr@[a+h]) in
  prefix (x, 0, []);;
prefix_sum [10;4;8;2];;
```

(\* 実行結果 \*)

```
val prefix_sum : int list -> int list = <fun>
- : int list = [10; 14; 22; 24]
```