

Homework 1

SNU 4190.310, 2025 봄

이 광근

Due: 3/15, 24:00

Exercise 1 “합쳐요”

큰 순서대로(descending order) 나열된 정수 리스트 두개를 받아서 하나의 순서 리스트로 만드는 함수

```
merge: int list * int list -> int list
```

를 정의하세요. 리스트에는 같은 정수가 반복해서 들어있을 수 있습니다. □

Exercise 2 “씨그마”

우리가 중고등 수학시간에 슬하게 썼던 다음의 “씨그마”를 OCaml로 정의하세요:

$$\sum_{n=a}^b f(n)$$

씨그마의 타입은

```
sigma : int * int * (int -> int) -> int.
```

즉, $\text{sigma}(a,b,f)$ 로 표현하면 $\sum_{n=a}^b f(n)$ 과 같도록. □

Exercise 3 “반복기”

다음함수 `iter`를 정의하세요:

$$\text{iter}(n, f) = \underbrace{f \circ \dots \circ f}_n$$

이 때, $n = 0$ 이면 아무일을 하지 않는(identity) 함수를 내 놓고, 양수이면 그 만큼 f 를 반복해서 적용하는 함수를 내 놓습니다. 그래서,

```
iter(n, function x -> 2*x) 0
```

은 $2 \times n$ 을 내 놓게됩니다. □

Exercise 4 “대진표 스트링”

일반적으로 게임 대진표는 완전 두갈래 가지구조(complete binary tree)입니다. 2025 월드컵 팀들과 그 대진표를 다음과 같이 정의했습니다:

```
type team = Korea | France | Usa | Brazil | Japan | Nigeria | Cameroon
           | Poland | Portugal | Italy | Germany | Norway | Sweden | England
           | Argentina
type tourna = LEAF of team
            | NODE of tourna * tourna
```

`tourna`를 받아서 괄호를 이용한 1차원 스트링으로 변환해주는 함수 `parenize`를 작성하세요:

```
parenize: tourna -> string
```

예를들어,

```
parenize(NODE(NODE(LEAF Korea, LEAF Portugal), LEAF Brazil))
= "((Korea Portugal) Brazil)"
```

□

Exercise 5 “참거짓”

Propositional Logic 식들(formula)을 다음과 같이 정의했습니다:

```

type formula = TRUE
              | FALSE
              | NOT of formula
              | ANDALSO of formula * formula
              | ORELSE of formula * formula
              | IMPLY of formula * formula
              | LESS of expr * expr
and  expr = NUM of int
              | PLUS of expr * expr
              | MINUS of expr * expr

```

주어진 `formula`를 받아서 참값을 만들어내는 함수 `eval`

```
eval : formula → bool
```

를 정의하세요. □

Exercise 6 “ k -진수”

일반적으로 k -진수($k > 1$)는 다음과 같이 표현한다.

$$d_0 \cdots d_n$$

여기서

$$\forall d_i \in \{0, \dots, k-1\}.$$

그리고 “ $d_0 \cdots d_n$ ”은 크기가

$$d_0 \times k^0 + \cdots + d_n \times k^n$$

인 정수를 표현한다.

이것을 살짝 확장해서 “ k -진수”를 다음과 같이 정의해보자. 표현은

$$d_0 \cdots d_n$$

여기서

$$\forall d_i \in \{1 - k, \dots, 0\} \cup \{0, \dots, k - 1\}.$$

그리고 “ $d_0 \dots d_n$ ”은 크기가

$$d_0 \times k^0 + \dots + d_n \times k^n$$

인 정수를 표현한다.

예를 들어, 3진수의 경우를 생각하자. 베이스가 $\{-2, -1, 0, 1, 2\}$ 이 되겠다. 0이 0을, +가 1을, -가 -1을, *가 2를, /이 -2를 표현한다고 하면, +는 1을, 0/+는 3을, +-는 -2을, +*0-는 -20인 정수를 표현한다.

OCaml로 3진수라는 타입을 다음과 같이 정의했다:

```
type crazy3 = NIL | ZERO of crazy3
             | ONE of crazy3 | MONE of crazy3
             | TWO of crazy3 | MTWO of crazy3
```

예를 들어, 0+/-은

```
ZERO(ONE(MTWO NIL))
```

로 표현된다.

위와 같이 표현되는 3진수를 받아서 그것의 값을 계산하는 함수 `crazy3val`을 정의하세요.

```
crazy3val: crazy3 -> int.
```

□

Exercise 7 “3진수의 합”

두 3진수를 받아서 3진수의 합에 해당하는 3진수를 내어놓는 함수 `crazy3add`를 정의하세요.

```
crazy3add: crazy3 * crazy3 -> crazy3
```

위의 `crazy3add`는 다음의 성질이 만족되어야 한다: 임의의 3진수 z 과 z' 에 대해서

$$\text{crazy3val}(\text{crazy3add}(z, z')) = \text{crazy3val}(z) + \text{crazy3val}(z').$$

□