

# Homework 1

SNU 4190.310, 2019 가을

이 광근

**Due: 9/14, 24:00**

## Exercise 1 “씨그마”

우리가 중고등 수학시간에 술하게 썼던 다음의 “씨그마”를 OCaml로 정의 하세요:

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

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 2 “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 만

함수  $f$ 를 반복해서 적용하는 함수를 내놓습니다. 그래서,

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

은  $2 \times n$ 을 내놓게됩니다. □

### Exercise 3 “대진표 스트링”

일반적으로 게임 대진표는 완전한 이진 나무구조(complete binary tree)입니다. 2022 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
```

tourna를 받아서 괄호를 이용한 1차원 스트링으로 변환해주는 함수 parenize를 작성하세요:

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

예를들어,

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

□

### Exercise 4 “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 formula를 받아서 참값을 만들어내는 함수 eval

```
eval : formula → bool
```

를 정의하세요. □

### Exercise 5 “ $k$ -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, 2진수의 경우를 생각하자. 베이스가  $\{-1, 0, 1\}$ 이 되겠다. 0이 0을, +가 1을 -가 -1을 표현한다고 하면, + 는 1을, +0+는 5를, +-는 -1을, +-0-는 -9인 정수를 표현한다.

OCaml로 2진수라는 타입을 다음과 같이 정의했다:

```
type crazy2 = NIL | ZERO of crazy2 | ONE of crazy2 | MONE of crazy2
```

예를 들어, 0+-은

```
ZERO(ONE(MONE NIL))
```

로 표현된다.

위와 같이 표현되는 2진수를 받아서 그것의 값을 계산하는 함수 `crazy2val`을 정의하세요.

```
crazy2val: crazy2 -> int.
```

□

### Exercise 6 “진수의 합”

두 2진수를 받아서 2진수의 합에 해당하는 2진수를 내어놓는 함수 `crazy2add`를 정의하세요.

```
crazy2add: crazy2 * crazy2 -> crazy2
```

위의 `crazy2add`는 다음의 성질이 만족되어야 한다: 임의의 2진수  $z$  과  $z'$ 에 대해서

$$\text{crazy2val} (\text{crazy2add}(z, z')) = \text{crazy2val}(z) + \text{crazy2val}(z').$$

□

### Exercise 7 “계산해줘!”

정수식들의 구조를 다음의 타입으로 정의했습니다:

```
type expr = NUM of int
          | PLUS of expr * expr
          | MINUS of expr * expr
          | MULT of expr * expr
          | DIVIDE of expr * expr
          | MAX of expr list
```

주어진 `expr`를 받아서 정수값을 만들어내는 함수 `eval`

```
eval: expr -> int
```

를 정의하세요. 이때, `MAX [NUM 1, NUM 3, NUM 2] = 3`, 즉 `MAX`는 정수식 리스트에서 가장 큰 정수를 찾아내는 정수식입니다. 빈 리스트의 경우는 0을 의미하는 정수식입니다. □