

ML演習第2回

課題

末永, 遠藤, 大山

課題 9 (special)¹

“文字集合 Σ 上の正規表現” と, 正規表現が表す文字列の集合は

- \emptyset は Σ 上の正規表現であり, 空集合を表す.
- ϵ は Σ 上の正規表現であり, 空列のみからなる集合 $\{\epsilon\}$ を表す.
- $s \in \Sigma$ ならば s は Σ 上の正規表現であり $\{s\}$ を表す.
- r と s がそれぞれ文字列の集合 R と S を表す Σ 上の正規表現であるとき, $r | s$ と rs と r^* とは Σ 上の正規表現であり, それぞれ $R \cup S$, $\{xy \mid x \in R, y \in S\}$, $\{x^n \mid x \in R, n \geq 0\}$ を表す².

のように定義される.

1. 正規表現を表すデータ型 `regex` を定義せよ. 文字集合について `polymorphic` な定義にすること.
2. 文字集合のリストで文字列³を表すことにする. 文字列と正規表現を受け取り, その文字列が正規表現にマッチするかどうかを判定する関数 `match_regex : 'a regex -> 'a list -> bool` を書け. 効率は特に気にする必要はない. (気にしてもよい)

¹special 課題は無理に解く必要はない. 他の授業や実験の課題が終わって本当に余力のある人だけ, 息抜きのもりでチャレンジして欲しい.

² x^n で文字列 x を n 回繰り返した文字列を表す.

³この設問中で言う文字列は, OCaml のデータ型としての文字列ではなく, 形式言語理論で言うところの文字列である.

ヒント データ型の定義ができれば, 実はそんなに難しくない. データ型は正規表現の定義に沿ってバリエーション型で定義すればよい.

```
type 'a regex =  
  Emptyset (* 空集合 *)  
| Null (* 空列 *)  
| Token of 'a (* 一文字の文字列 *)  
| Repeat of 'a regex (* 繰り返し *)  
...
```

この定義を使えば

```
# match_regex (Token 1) [2];;  
- : bool = false  
# match_regex (Repeat(Token 2)) [2; 2; 2];;  
- : bool = true  
# match_regex (Repeat(Token 2)) [2; 2; 1];;  
- : bool = false  
# match_regex (Repeat(Token 2)) [];;  
- : bool = true
```

となるはずである.