

EOPL Pset 4

Alireza Habibzadeh 99109393

5

این تابعی که به دنبال آن هستیم همان `occurs-free?` است که قبلا نیز به آن برخورد کرده بودیم:

```
(define occurs-free?
  (λ (var exp)
    (cond
      ((symbol? exp) (eqv? var exp))
      ((eqv? (car exp) 'λ)
       (and
        (not (eqv? var (car (cadr exp))))
        (occurs-free? var (caddr exp))))
      (else
       (or
        (occurs-free? var (car exp))
        (occurs-free? var (cadr exp)))))))
```

حال باید متغیرهای `free` (نسبت به آن `procedure`) را فیلتر کنیم (یعنی فقط آن‌ها را لحاظ کنیم) پس `filter` را با تابع چکر `occurs-free?` بر روی `env` صدا می‌زنیم. البته `env` باید به صورت لیستی (و نه تابعی) پیاده شده باشد. برای حالت `env` تابعی باید خودمان تابع `filter` را (البته با یک اسم دیگر) پیاده کنیم.

شکل `datatype` که بدیهتا تغییری نمی‌کند. تابع `value-of`:

```
(define value-of
  (λ (exp env)
    (cases expression exp
      (proc-exp (var body)
        (proc-val
         (procedure var body
                     (filter
                      (λ (var val) (occurs-free? var body))
                      env))))
      ; .
      ; .
      ; . other expressions
    )))
```