

```

(require-library "pretty.ss")

(load-extension "MoreStrings debug.so")

(define (translatex s rx transform . procs)
  (define len (string-length s))
  (define o (open-output-string))
  (define (translate start)
    (define (search)
      (define positions (regexp-match-positions rx s start))
      (define (write-match)
        (define (get-string position)
          (define pos1 (car position))
          (define pos2 (cdr position))
          (substring s pos1 pos2))
        (define strings (map get-string (cdr positions)))
        (define (apply1 proc arg)
          (proc arg))
        (define transformation
          (apply transform (map apply1 procs strings)))
        (define pos1 (caar positions))
        (define pos2 (cdar positions))
        (begin
          (when (< start pos1)
            (display (substring s start pos1) o))
            (display transformation o)
            (translate pos2))))
      (define (write-rest)
        (display (substring s start len) o))
      (if positions
        (write-match)
        (write-rest)))
    (begin
      (when (< start len)
        (search))
      (get-output-string o)))
  (translate 0))

(define (translate s r transform . procs)
  (define rx (regexp r))
  (apply transllatex s rx transform procs))

(define s "proc() begin mAbc = inDef; end")

```

```
(define r "(ini|ol|ml|out|s)([A-Z])([a-zA-Z0-9]+)")
(define rx (regexp r))
```

```
(define (test)
  (begin
    (pretty-print (regexp-match rx s))
    (pretty-print (regexp-match-positions rx s))
    (pretty-print (regexp-match rx s 17))
    (pretty-print (regexp-match-positions rx s 17))))
```

```
(define (improve s)
  (define (transform x y z)
    (string-append x "_" y z))
  (define (proc1 x) x)
  (define (proc2 y) (lowercase-string y))
  (define (proc3 z) z)
  (translate s rx transform proc1 proc2 proc3))
```

```
(improve s)
```

```
(define (translate-file input-path output-path)
  (define i (open-input-file input-path))
  (define o (open-output-file output-path 'truncate/replace))
  (define (convert)
    (define line (read-line i 'any))
    (if (eof-object? line)
        (pretty-print (list 'done output-path))
        (begin
          (display (improve line) o)
          (newline o)
          (convert))))))
(begin
  (convert)
  (close-input-port i)
  (close-output-port o)))
```

```
(translate-file "LPane.cp" "LPane.cp+")
(translate-file "LPane.h" "LPane.h+")
```

```
; This is just an example.
; It won't be suitable for every C++ file.
; For example, SetTextTraitsID will be replaced
; with SetTextTraits_iD.
```