Abstract Syntax and Its Representation

Exercise 2.28

```
(define unparse-lc-exp
  (lambda (exp)
    (cases lc-exp exp
      (var-exp (var)
               (symbol->string var))
      (lambda-exp (bound-var body)
                  (string-append
                   (string-append "(lambda ("
                                  (symbol->string bound-var))
                   (string-append ") "
                                  (unparse-lc-exp body))))
      (app-exp (rator rand)
               (string-append
                (string-append "("
                               (unparse-lc-exp rator))
                (string-append
                 (string-append " "
                                (unparse-lc-exp rand))
                 ")")))))))
```