Example Program

```
(define zip
  (lambda (ls1 ls2)
    (if (null? ls1)
        '()
        (cons (cons (car ls1)
                    (list (car ls2)))
              (zip (cdr ls1) (cdr ls2))))))
```

Context Choice Weight triplets

(app-rand*	cdr	2)
(app-rator	var	1)
(car	var	2)
(cdr	var	2)
(cons-e1	car	1)
(cons-e1	cons	1)
(cons-e2	app	1)
(cons-e2	list	1)
(if-alt	cons	1)
(if-conseq	quoted-datum	1)
(if-test	null?	1)
(lambda	if	1)
(letrec-rhs	lambda	1)
(list	car	1)
(null?	var	1)
(top-level	letrec	1)

Synthesis guided by triplets

```
(define append
  (lambda (l s)
    (if ???
        ???
        ???)))
```

```
(define append
  (lambda (l s)
    (if (null? ???)
        ???
        ???)))
```