Lesson Plan - 6.001 SP04 - recitation 25

stacks and recursion, procedure call

problem 1

input: arg0 arg1
output: result
modifies: arg1 result

problem 2

input: x
output: val
modifies: x val

problem 3

```
aexpb
 (save arg0)
 (save continue)
 (assign arg0 (const 2.71828))
 (assign continue (label aexpb-after))
 (goto (label expt))
aexpb-after
 (restore continue)
 (restore arg0)
 (assign result (op *) (reg arg0) (reg result))
 (goto (reg continue))
```

problem 4

```
list-copy
 (test (op null?) (reg arg0))
 (branch (label list-copy-done))
 (save arg0)
 (save continue)
 (assign arg0 (op cdr) (reg arg0))
 (assign continue (label list-copy-after))
 (goto (label list-copy))
list-copy-after
 (restore continue)
 (restore arg0)
 (assign tmp (op car) (reg arg0))
 (assign result (op cons) (reg tmp) (reg result))
 (goto (reg continue))
list-copy-done
 (assign result (const ()))
 (goto (reg continue))
```

problem 5

```
list-ref
 (test (op =) (reg arg1) (const 0))
 (branch (label list-ref-done))
 (save continue)
 (assign arg0 (op cdr) (reg arg0))
 (assign continue (label list-ref-after))
 (goto (label list-ref))
list-ref-after
 (restore continue)
 (goto (reg continue))
list-ref-done
 (assign result (op car) (reg arg0))
 (goto (reg continue))
```

problem 6

```
sum-of-exps
 (assign result (const 0))
sum-of-exps-top
 (test (op null?) (reg arg0))
 (branch (reg continue))
 (save arg0)
 (save continue)
 (save result)
 (assign arg1 (op car) (reg arg0))
 (assign arg0 (const 2.71828))
 (assign continue (label sum-of-exps-after-expt))
 (goto (label expt))
sum-of-exps-after-expt
 (assign tmp (reg result))
 (restore result)
 (restore continue)
 (restore arg0)
 (assign result (op +) (reg result) (reg tmp))
 (assign arg (op cdr) (reg arg0))
 (goto (label sum-of-exps-top))
```