Pseudocode Reference Guide

Saddleback College

TERMINAL

```
BEGIN
    <statement(s)>
END
```

Be sure to indicate if this is used for BEGIN, END, or RETURN

INPUT/OUTPUT STATEMENTS

INPUT

```
PROMPT input_variable
```

OUTPUT

```
OUTPUT identifier, expression, or literal
```

PROCESSING/ASSIGN

```
ASSIGN variable = literal
```

Use assign when assigning a literal to a variable

```
CALCULATE or CALC variable = expression
```

Use CALC or CALCULATE when assigning an expression to a variabl

REPETITION STATEMENTS

FOR LOOP (known # of iterations)

```
FOR lcv = initial_value TO end_value (loop x times)
    <loop_statement(s)>
END FOR
```

WHILE LOOP (unknown # of iterations)

```
WHILE boolean_expression
    <loop_statement(s)>
END WHILE
```

DO-WHILE LOOP (post-test loop)

```
DO
    <loop_statement(s)>
WHILE boolean_expression
```

FINAL NOTES

- ALWAYS include a variable list with your pseudocode
- CAPITALIZE action words as shown
- don't forget to indent as shown
- always have BEGIN and END
  indent the statements between them

SELECTION STATEMENTS

IF-THEN (one-way)

```
IF boolean_expression THEN
    <true_statement(s)>
END IF
```

Note: There are no false instructions.

IF-THEN-ELSE (two-way)

```
IF boolean_expression THEN
    <true_statement(s)>
ELSE
    <false_statement(s)>
END IF
```

IF-THEN-ELSE-IF (multi-way)

```
IF boolean_expression_1 THEN
    <true_statement(s)_ 1>
ELSE IF boolean_expression_2
    <true_statement(s)_ 2>
.
.
.
ELSE
    <false_statement(s)>
END IF
```

SWITCH STATEMENT (multi-way)

```
SWITCH expression
    CASE <constant_expr_1> : <statement(s)_1>

    CASE <constant_expr_2> : <statement(s)_2>


    .
    .
    .
    DEFAULT : <default_statement(s)>
                                break;
END SWITCH
```