

## CST8110: Introduction to Computing Using Program Definition Language to Solve a Problem

GET	Gets input values from the user. Stores those input values in some variable.
PUT	Displays heading or prompt (surrounded by double quotes). Displays contents of some variable.
←	Assignment: Take the value on the right side of the arrow and assign it to the variable on the left side of the arrow. Note: Word will automatically generate this single-character symbol when you type the 3-character sequence <--
IF-ENDIF	This is a <i>selection</i> construct. The IF tests the state of some variable. All instructions between the IF and ENDIF are performed if the condition is true.
IF-ELSE-ENDIF	This is another <i>selection</i> construct. The IF tests the state of some variable. All instructions between the IF and ELSE are performed if the condition is true. All instructions between the ELSE and ENDIF are performed if the condition is false.
WHILE-ENDWHILE	This is a repetition (looping) construct. The WHILE tests the state of some variable. All instructions between the WHILE and ENDWHILE are performed if the condition is true. At the end of one iteration of the loop, the WHILE is tested again, and may repeat the instructions. (Note: This suggests that something inside the loop must be changing the variable being tested.
Arithmetic Operators: + - * /	The four traditional arithmetic operators are used to: <i>add</i> , <i>subtract</i> , <i>multiply</i> and <i>divide</i> . Multiply is performed using the asterisk *. Divide is performed using the slash /
Relational Operators: < <= > >= == !=	When defining the test conditions in your IF instructions and your WHILE instructions, you will commonly use these symbols. For example: <i>IF Age &gt;= 16</i>
Boolean Operators: AND OR	When you have multiple parts to a test, you can combine them with the AND OR operators. For example: <i>IF Number &lt; 0 OR Number &gt; 100</i>

I purchase an item. The sales clerk tells me the amount owed. I give a single bill which is large enough to pay the full amount. Give me exact change.

### Problem Statement (Assumptions)

- Amount owed is less than \$100.
- Pay with a single \$100 bill.
- Count change by giving largest denomination first and working backward through available denominations. (The store refuses to handle anything larger than a \$20 bill.)
  - \$20
  - \$10
  - \$5
  - \$2
  - \$1
  - .25
  - .10
  - .05
  - .01
- Process assembling change by counting one instance of a monetary token at a time (for example, with a purchase of \$37.43, I would count out change of one \$20 bill, then another \$20 bill, and then another \$20 bill, in total 3 - \$20 bills, but only one at a time).
- Processing will be done with loops to perform the counting of change.