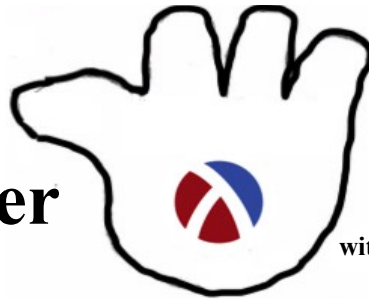


introductory

# C++



## design recipe helper

with apologies to Betty Crocker

**1.** // signature:  new function NAME:  TYPE(S) of expression(s) it EXPECTS:  TYPE of expression it RETURNS:

**2.** /\* purpose: DESCRIBE what it EXPECTS:  expects  and returns DESCRIBE what it RETURNS:  \* /

**3.** function header: for EACH expression it expects, give a PARAMETER TYPE & descriptive NAME  
 TYPE function RETURNS, from 1. function NAME, from 1. ( param TYPE param NAME param TYPE param NAME  
  ,   ,  
 param TYPE param NAME param TYPE param NAME )

**4.** tests: function NAME from 1. : (  ) for each PARAMETER, give an example ARGUMENT value, SEPARATED BY COMMAS  
 == (  )  
 if works, give expression that SHOULD be equal to: (  )  
 -----  
 ( function NAME from 1. : (  ) for each PARAM, give an ex DIFFERENT ARGUMENT val, SEP'D BY COMMAS  
 == (  )  
 if works, give expression that SHOULD be equal to: (  )

**5.** complete function body:  
 function header from 3. :   (  )  
 { return  ;  
 }

**6.** add tests to testing main () function,  
 then COMPILE -- any SYNTAX errors? then RUN -- do tests pass?  
 if so: YOU'RE DONE! if not: DEBUG 3., 4., 5. until they do!