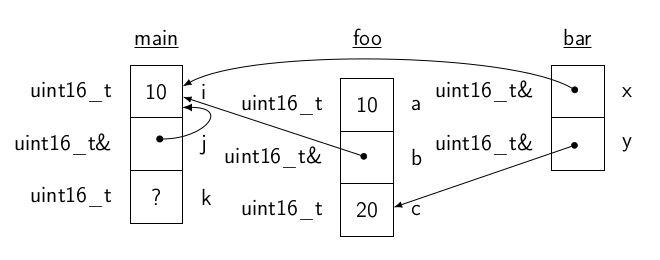
**Assignment 124: Memory**

Last modified: 14 January 2022



Write a C++ program that declares variables and allocates memory to initially match the diagram of memory above. The program must have exactly three functions, main, foo, and bar.

The function main must initialize local variables i and j as shown in the diagram, and then must call the function foo, storing the return value of foo in the local variable k.

The function foo must have exactly two formal parameters named a and b. Write statements in foo so that the value of a (which is initially 10) is divided by 2, the value of b is incremented by 1, and the local variable c is initialized to 20. The function foo must call the function bar, passing parameters as shown in the diagram. After calling bar, add a statement to foo so that the return value is the sum of a, b, and c; this value is returned to the calling scope.

The function bar, which does not have a return value, must have exactly two formal parameters named x and y. Add statements to bar to double the value of x and to increment the value of y by three.

Immediately before bar terminates, add statements to output the final values of x and y, labeled, one value per line. Immediately before foo returns, add statements to output the final values of a, b, and c, labeled, one value per line. Immediately before main returns, add statements to output the final values of i, j, and k, labeled, one value per line.

A run of your program should produce output that looks exactly like this, except that these values as shown are not necessarily correct:

$ ./program

x: 12

y: 34

a: 56

b: 78

c: 91

i: 23

j: 45

k: 67

A significant part of this assignment is for you to verify that the values printed by your program are correct, based on the structure of the program matching the diagram above.

For this assignment, you should suspend the style guide rules about magic numbers, single-letter variable names, and Doxygen comments on functions. You should hard-code the indicated values and use the single-letter variable names I have indicated. Other than this, make sure your code adheres to the [course coding standards](https://borax.truman.edu/310/coding_rules.html). Of particular note: spaces on both sides of binary operators and after commas, no spaces between a unary operator and its argument, make sure output is terminated with a newline, and no tab characters in the source code.

Remember that according to academic honesty policy stated in the course syllabus, you are not allowed to share or discuss any of the code of this assignment, nor are you allowed to discuss what the values of the final output should be, with anyone other than me.

By 3pm on Monday, 24 January, submit the C++ source code file to the [homework submission](https://borax.truman.edu/310/submit.php) page.