higher memory

addresses

lower memory

addresses

Var name value addr



// Local vars



// on the Stack



int x;

int a[3];



int \*ptr;



0x8000



Use the same variable name in the "Same as" expressions. For example, if the expression uses ptr, you cannot use the array name a in the equivalent expression.

Generally, if the expression involves a pointer access, write the equivalent using an array access, and vice versa.



ptr = &a[0]; // Same as \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



\*ptr = 42; // Same as \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



\*(a + 1) = 59; // Same as \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

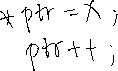


ptr[2] = 77; // Same as \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



x = 82;

\*ptr++ = x; // Equivalent as two statements \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

x = ++\*ptr; // Equivalent as two statements \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



x = 99;

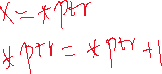
\*++ptr = x; // Equivalent as two statements \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

x = 5;

x = (\*ptr)++; // Equivalent as two statements \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

x = \*(ptr - 2) + 5; // Same as \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

