8.7 a. True

8.7 b. False

8.8 a. unsigned int values[5] = {2,4,6,8,10};

8.8 b. unsigned int \*vPtr;

8.8 c. for(int i=0, i<5, i++)

cout(<< values[i]);

8.8 d. vPtr = &values[1]

8.8 e. for(int I = 0, i<5, i++)

Cout(<<\*vPtr);

vPtr++;

8.8 f. for(int I = 0, i<5, i++)

Cout(<<(\*(values+i));

8.8 g. for(int i = 0, i<5, i++)

Cout(<<vPtr[i]);

8.8 h. values[4];

\*(values+4);

vPtr[4];

\*(vPtr + 4);

8.8 i. 1002506;

8

8.8 j. 1002500;

2;

8.11 a. number missing an \*.

8.11 b. realPtr and integerPtr are different types. Must be same type.

8.11 c. x doesn’t point to anything. Must be initialized to point to a value.

8.11.d Missing \* before s in for statement

8.11 e. numPtr is a null pointer, and genericPtr gets the address of it, and so you can perform the arithmetic.

8.11 f. xPtr missing an \*.

9.17. Yes, because the first constructor will always be used, and the second one will do nothing.