Lab 12: STL Sets and Maps

Create a **new project** and implement the **non-member functions** listed below. Add the specified testing cases in the **main** function to test your functions.

- **Function countClumps**
  - **Parameter:** An **STL multiset** of integers
  - Say that a "clump" in a sequence is a series of 2 or more adjacent elements of the same value. Return the number of clumps in the given multiset.

TESTING CASES	OUTPUT
{1, 2, 2, 3, 4, 4}	2
{1, 1, 2, 2, 2}	2
{3, 3, 3, 3, 3}	1
{1, 2, 3}	0
{2, 2, 6, 6, 6, 7, 8, 8, 9, 9}	4
{1, 2, 3, 4, 5, 6, 7, 8, 9, 9}	1
{1, 3, 5, 7, 7, 8, 9}	1
{}	0

- Function **linearIn**
  - **Parameters:** A **non-empty STL multimap** of integers
  - The function returns true if all the numbers in the **second** variable of the pair appear as the **first** variable of any other pair.

TESTING CASES	OUTPUT
(3,3), (4,4), (5,1), (6,2), (7,3), (7,5)	false
(1,2), (2,6), (4,2), (4,6), (6,4)	true
(2,4), (2,3), (4,6), (4,7), (6,5), (6,2)	false
(3,6), (4,6), (5,6), (6,6)	true
(3,3), (3,3), (3,3), (3,3), (3,4)	false
(3,3), (3,3), (3,3), (3,4), (4,3)	true

- Function **multiples**
  - **Parameters:** An **empty STL set** and an **integer**
  - The function populates the set with 10 multiples of the given integer, except for the values that end with the integer itself.

TESTING CASES	OUTPUT
2	{4, 6, 8, 10, 14, 16, 18, 20, 24, 26}
3	{6, 9, 12, 15, 18, 21, 24, 27, 30, 36}
5	{10, 20, 30, 40, 50, 60, 70, 80, 90, 100}

After implementing your functions, re-visit them and see if there is a way to make them **more efficient**.