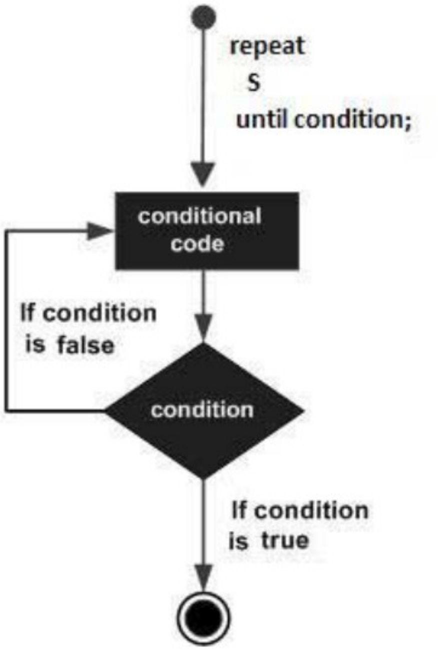
Implement the following methods, etc. Use the provided data model when needed, it is related with films, actors/actresses and awards. Non functional constructions will be penalized except for trivial tasks such as showing the elements in a collection. Submit a single self contained solution with:
The provided .cs file including the requested methods related with LINQ.
Your IEnumerable extension methods with the requested additional method.
Your IEnumerable generic list.
A console app with the closure exercise (c&p the WhileLoop closure), requested methods calls and any additionally implemented code.
Use comments to label the code for each exercise.

1. Using the WhileLoop closure emulating a while, implement the Pascal repeat until control structure.



Use that closure to show numbers from 0 to 9, both included,

2. Using two generic lists (your own list, not a C# one), generate two lists holding the same number of double (3 or 4 will do). Using your own extension methods, implement a method that computes the euclidean distance between two double vectors. Call this method using the mentioned lists as parameters.

3. Add the method ReduceRange to your extension methods. This method takes as parameters two int that define the index range for the data that will be reduced. For instance, if the collections holds 3,4,2,4,5,6 and the parameters are 1 and 3, the reduce operation will be applied only to 4, 2 and 4.
You can use foreach here but it is not needed. Invoke the method from main adding any collection of numbers between a given index range.

4. Using LINQ (Count is forbidden) implement a method that returns the number of films with a given format (definition) and genre. Both parameters are strings. Do not forget to invoke from Main.

5. Using LINQ (no Max allowed, but you can freely use Count) most awarded prize. Invoke in main.