OLLSCOIL NA hÉIREANN, CORCAIGH
THE NATIONAL UNIVERSITY OF IRELAND, CORK

COLÁISTE NA hOLLSCOILE, CORCAIGH
UNIVERSITY COLLEGE, CORK

SUMMER EXAMINATIONS 2014

CS 6501 : Programming for Bioscientists I

Professor Ian Gent
Professor Barry O'Sullivan
Dr Joseph Manning

Answer all questions

$1\frac{1}{2}$ Hours

Please ensure that you have the correct exam paper

DO NOT TURN THE PAGE UNTIL INSTRUCTED TO DO SO

(*25%*) 1. Write a Python function `Firsts( s )` to return the string formed from the first occurrence of each item in string `s`. For example:

```
Firsts( "mississippi" )  ⇒  "misp"
Firsts( "abcdefg"     )  ⇒  "abcdefg"
Firsts( "aaaaaaaaa"   )  ⇒  "a"
Firsts( ""            )  ⇒  ""
```

(*25%*) 2. Write a Python function `IsStairs( s )` to test if the numeric list `s` is a stairs (a *stairs* is a list of at least two numbers where *either* each number is one greater than the previous number *or* each number is one smaller than the previous number). For example:

```
IsStairs( [2,3,4,5] )  ⇒  True
IsStairs( [8,7,6]   )  ⇒  True
IsStairs( [2,3,5]   )  ⇒  False
IsStairs( [2,3,2]   )  ⇒  False
IsStairs( [4]       )  ⇒  False
```

(*25%*) 3. Write a Python function `LineCount( filenames )` to take a list `filenames` of file names and write out each file name and the number of lines in that file, finally writing out the total number of lines in all files. Assume that each file name has at most 20 characters, and that the total number of lines is at most 999,999. If any of the files cannot be read, then issue an appropriate error message and otherwise ignore that file. For example:

```
LineCount( ["Jingle-Bells","Bogus-File","Fields-of-Athenry"] ) ⇒
   Jingle-Bells            28
   Bogus-File                 cannot read file
   Fields-of-Athenry       40
   TOTAL                   68
```

(*25%*) 4. a) Write a Python function `Counts( s )` to return a dictionary, in which each key is a distinct item in sequence `s` and the corresponding value is the number of times that the item occurs in `s`; for efficiency, this function should inspect each item in `s` only *once*. For example:

```
Counts( "ccabbadcac" )   ⇒  {"a":3, "c":4, "b":2, "d":1}
Counts( [4,7,4,7,4] )    ⇒  {7:2, 4:3}
```

(recall that the order in which keys appear in a dictionary is irrelevant) (*10%*)

b) Write a Python function `MostFrequent( s )` to return an item in the non-empty sequence `s` which occurs at least as many times as every other such item. (*10%*)

c) Write a Python function `AreAnagrams( s1, s2 )` to test if sequences `s1` and `s2` are anagrams of one another (*anagrams* contain the same overall collection of items, but perhaps in different orders). For example:

```
AreAnagrams( "aabccc", "cabcac" )  ⇒  True
AreAnagrams( "aabccc", "aabbcc" )  ⇒  False
AreAnagrams( "aabccc", "aabc"   )  ⇒  False
```

(*5%*)