**MCS 3312 Analysis of Algorithms – Fall 2017**

Homework Assignment #1

Problem 1:

Part A:

Using python, implement the methods specified in Cantor’s diagonalization proof. Your python script should query the user for an arbitrary number of iterations, and then report the resulting enumeration.

**Pseudo Code for Cantor’s diagonalization proof up to iteration *n*:**

*n* ← get from user

*digitSet* ← [*w*, *m*]

*Table* ← ∅

for *i* ← 2 to *n*

*Table* ← *Table* ⊗ *digitSet*

*print*(*Table*)

*O*(*c*)

*O*(*c*)

*O*(*c*)

*O*(*n*2*n*) ≥ *O*(*n*2*i*)

*O*(2*i*)

*O*(2*n*)

*f*(*n*) = *O*(*c*) + *O*(*c*) + *O*(*c*) + *O*(*n*2*n*) + *O*(2*n*) = *O*(*n*2*n* + 2*n* 3*c*) ≅ *O*(*n*2*n*)

Part B:

Using the substitution method for recursions, compute the solution to Cantor’s diagonalization proof.

*S*(*n*) := the size of the table

*n* := the number of iterations

*s*(*n*) := the size of the current table

*r* := the radix

*S*(*n*)(*n =* 0) := *rs*(*n*) = (2)(20) = 2

*S*(*n*)(*n =* 1) := *r*(*rs*(*n*)) = *r*2*s*(*n*)

*S*(*n*)(*n =* 2) := *r*(*r*2*s*(*n*)) = *r*3*s*(*n*)

*S*(*n*)(*n =* 3) := *r*(*r*3*s*(*n*)) = *r*4*s*(*n*)

*S*(*n*)(*n =* 4) := *r*(*r*4*s*(*n*)) = *r*5*s*(*n*)

* So, the growth of the table is *O*(2*n*).

Problem 2:

Part A:

Using python, implement the method for enumerating the set of all pairs.

Part B:

What is the geometric difference between the enumeration in Problem 1A and Problem 2A?

Problem 3:

Part A:

Using python, develop a divide and conquer method to compute the enumeration of the set of all reals (*i.e.*, Cantor’s diagonalization proof) for a given iteration.

**Pseudo Code for Cantor’s diagonalization proof up to iteration *n* (divide and conquer):**

InitializeSetOfReals(*A*, *n*)

|  |  |
| --- | --- |
| *A*[*n*] ← ∅ | *O*(*c*) |
| *q* ← log2 *n* | *O*(*c*) |
| GenerateSetOfReals(*A*, *1*, 2*q* − 1, ‘*w*’) | *O*(0.5*n* log *n*) |
| GenerateSetOfReals(*A*, 2*q*, *n*, ‘*m*’) | *O*(0.5*n* log *n*) |

GenerateSetOfReals(*A*, *p*, *r*, *s*)

|  |  |
| --- | --- |
| if *p* < *r* | *O*(*c*) |
| *q* ← log2 (*r* − *p*) | *O*(*c*) |
| *q* ← *p* + 2*q* | *O*(*c*) |
| GenerateSetOfReals(*A*, *p*, *q* − 1, ‘*w*’ + *s*) | *O*(log *n*) |
| GenerateSetOfReals(*A*, *q*, *r*, ‘*m*’ + *s*) | *O*(log *n*) |
| else if *p* = *r* | *O*(*c*) |
| *A*[*p*] = *s* | *O*(*c*) |

* the growth is *O*(*n* log *n*).

Part B;

Compute the growth in computational cost for the method in Problem 1A and the method that you developed for Problem 3A. (Show your work.)

Part C:

Did you run into any complications? If so, explain why.