Location-based Services for Surrounding Area with Myanmar Language on Mobile Devices

Myat Thiri Khine[1] and Myint Myint Sein[1]

[1] University of Computer Studies, Yangon
myatthirikhine@ucsy.edu.mm
myintucsy@gmail.com

Abstract. Location-based services provide the users who are not familiar with the region to find their desired location. There has been developed many location-based services on the mobile devices but the location-based services with Myanmar native is still poor. The pronunciation error is still occurred in existing application and it is still necessary to develop the efficient and effective one. Spatial index can be used to quickly retrieve the geo-information efficiently and effectively. Spatial index is one of the problems in the mobile devices because of the limited memory and low computation capacity. In this paper, a new index structure is proposed to retrieve the desired location based information for user's surrounding area with Myanmar language on the mobile devices. The proposed index structure is constructed by using Hilbert space filling curve and B-tree that combines the inverted file to reduce the searching time. Myanmar 3 is used as the Myanmar keyword in this paper to avoid the various sequence order of the character because the typing order of Myanmar language may be various form.

Keywords: Location-Based Service, Spatial Query, Hilbert Curve, B-tree, Index Structure, Inverted File.

1 Introduction

Although the effective Location-based services have been developed in the developed countries, it still lacks to develop the effective and efficient one in the developing countries as Myanmar. Location-based services application for Myanmar language has been developed but most of the applications mainly depend on the web services and still has the pronunciation error. So, it still needs to develop it on the mobile devices to easily search the location with Myanmar native language anywhere and anytime.

Spatial data which are data that contain a location mainly needed for Geographical Information System (GIS) whose information is related to geographic locations. It is stored in Geographical Information System (GIS) and the geo-information is accessed from the existing spatial data.

Geospatial database is used for both storage of Spatial and Non-Spatial attributes and uses Geographic Information System (GIS) to locate and retrieve data effectively and efficiently [20] [21]. It can be used for indexing data structures to quickly retrieve the spatial queries. Spatial query can take the geographical features depend on location or spatial relationship. The more searching time will be taken as the database size increases.

Many index structures [1-15] have been developed in recent years to quickly retrieve the geo-information. R-tree is mostly used to access the geo-information. R-tree with inverted file, namely the families of IR-tree [8, 9, 10, 11, 12, 13, 14], is mainly used for both spatial and textual search. R-tree is utilized for indexing the spatial data and inverted file is utilized for indexing the textual data. In R-tree, the unnecessary node might be visited and higher IO cost can be caused in the searching process as the data objects can be overlapping and covering each other [15].

Hybrid index structure that combines the K-d tree and the inverted file for spatial keyword search with the minimum IO costs and CPU costs has been proposed [1][2][3]. This hybrid index structure is considered for English keyword queries on the web services.

The nature of Myanmar language is complex. It has various types of characters such as consonants, medials, vowels, tones, etc. Myanmar word consists of one or more syllables that can contain one or more characters. Myanmar sentences do not have white space to specify words boundaries. Moreover, the sequence of the Myanmar characters is also important for matching the Myanmar word. The typing order of the Myanmar characters may be vary (pigeon, ခို → ခ + ိ + ု or ခ + ု + ိ). The sequence of characters must have the same order to match the word. In Myanmar 3 Unicode, the sequence of the character needs to be the correct order and the typing order of the character cannot be changed. For example, the sequences of လှေ(boat) syllable in Myanmar 3 Unicode is လ + ှ + ေ (101C 103E 1031) [11]. Therefore, Myanmar 3 is used in this paper to avoid the various sequence order of the character in Myanmar language.

In this paper, a new index structure is presented to effectively and efficiently retrieve the location within the given range with Myanmar language on the mobile devices. This index structure is constructed by using Hilbert space filling curve and B-tree with the inverted file and it can reduce the searching time.

The rest of the paper is arranged as follows. In Section 2, the related works is presented. Section 3 discusses the system design. Hilbert Curve is explained in section 4. Section 5 describes B-tree. The proposed index structure is described in section 6. The experimental result is shown in section 7. Section 8 concludes the paper with the directions for future work.

2 Related Works

There are many index structure has been developed to retrieve the geo-information. Among them, R-tree [12, 13, 8, 11, 10] is mostly utilized. Its variants are used for spatial index and inverted file is for text index. Both indices are combined based on the combination schemes [7]. The IR tree [8] constructs each R-tree nodes with a summary of the text content of the objects in the corresponding subtree. X. Cao, L.

Chen, G. Cong, C. S. Jensen, Q. Qu, A. Skovsgaard, D. Wu, and M. L. Yiu [6] proposed S2I index structure that is constructed based on R-tree and inverted file. I. D. Felipe, V. Hristidis, and N. Rishe [10] utilizes R*-tree for spatial index and inverted file for text index.

Hariharan et al. R. G¨obel, A. Henrich, R. Niemann, and D. Blank [11] presented the KR*-tree. This paper proposed a framework for GIR systems and focus on indexing strategies. T.Wang, G. Li, J. Feng [16] proposed a new index structure, spatial keyword R-tree, called SKR-Tree which extended from the R-tree with an R-tree node storing both spatial and keyword information.

The objects in [6] are stored differently depend on the document frequency and infrequency of the term. Cary et al, [13] proposed SKI which is constructed using R-tree with an inverted index that contains spatial references in posting lists. The posting list of term contains all its term as bitmaps rather than documents.

X. Chen, C. Zhang, B. Ge, W. Xiao presents TUR-tree and TUA-tree to accelerate the query process. Query processing algorithms are designed for the three queries in social network, aiming to explore temporal dimension in users, relationships and social activities [19]. Y. Tao and C. Sheng develop the spatial inverted index (SI-index) to answer nearest neighbor queries with keywords in real time [22].

X.Cao, G.Cong, Christian S. Jensen, Jun.J. Ng, BengC.Ooi, N.T. Phan, D. Wu [7] proposed a Web Object Retrieval System (SWORS) to efficiently access the spatial web objects that satisfy spatial keyword queries by using IR tree and inverted file for index. It supports two types of queries which are location aware top-k text retrieval (Lkt) query and spatial keyword group (SKG) query.

3 System Design

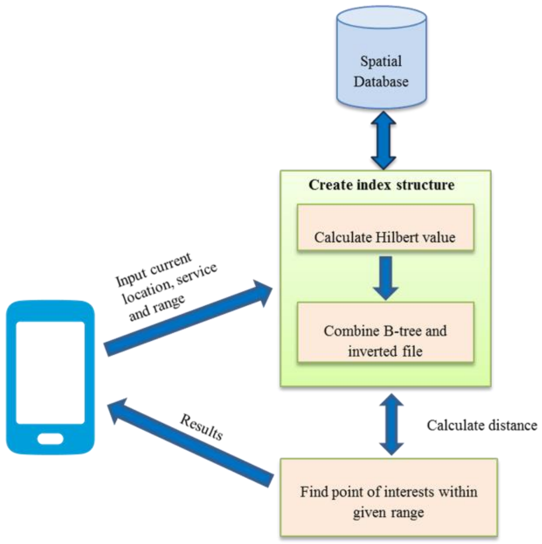

Fig. 1. System Architecture

This system is developed to provide the user to easily access the location with Myanmar languages on the mobile devices. The data with their location (latitude, longitude) and services are stored in the spatial database. In this system, the user current location, services and desired range are taken as input. The desired point of interest is searched in the proposed index structure based on the user current location, given services and user desired range. The distance is calculated by using the Euclidean distance. Then, all the point which is within the given range is returned to the user. The system architecture is shown in Fig. 1.

4 Hilbert Curve

The Hilbert Curve is space-filling curve which visits every point within a two dimensional space. The Hilbert curve is used for mapping an n-dimensional coordinate system to a 1-dimensional index. Fig. 2 shows the Hilbert curve orders one, two and three.

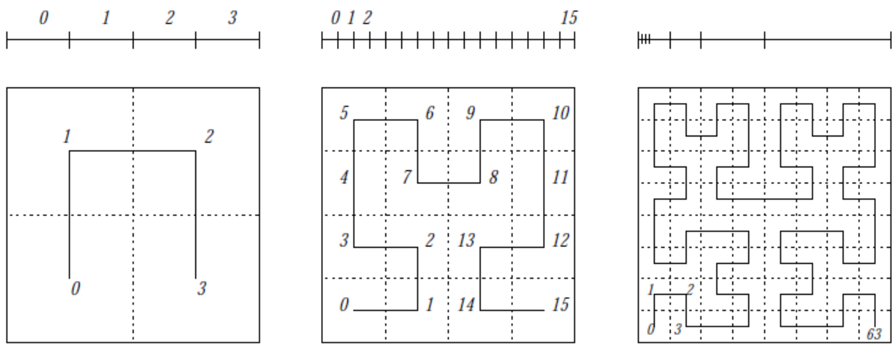

Fig. 2. Hilbert Curve Orders 1, 2, and 3 respectively.

The basic curve of the shape is an upside down “U”. Firstly, a square is separated into 4 quadrants. In the first-order, the curve is drawn through their center points. The quadrants are ordered such that any two adjacent points in the ordering share a common edge. The top vertices are replaced by the previous order, and the bottom vertices suffer a rotation. The bottom left vertex is rotated 90 degrees clockwise, and the bottom right rotates 90 degrees counter clockwise. The curve starts on the lower left corner and ends on the lower right corner [17].

5 B-Tree

B-tree is balanced search tree and is similar to red-black trees but they are better at minimizing disk I/O operations. B-tree or variants of B-trees are used in many database systems to store information. B-trees keep values in every node in the tree, and may use the same structure for all nodes. Unlike a binary-tree, each node of a b-tree may have a variable number of keys and children. The B-tree of order 2 can be seen in Fig. 3. In B-tree of order x, every node has at most x children and has at least x/2 children [23].

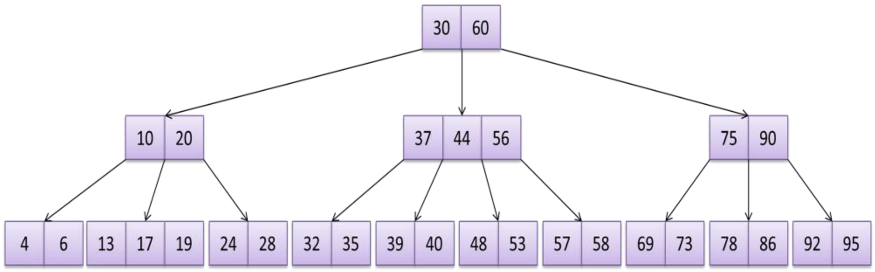

Fig. 3. B-Tree of Order 2

A B-tree R is a rooted tree whose root is R.root. In every node p, the number of keys p.n currently stored in node p. The p.n keys ($p.k_1, p.k_2, \ldots, p.k_{p.n}$) are stored in non-decreasing order so that $p.k_1 \leq p.k_2 \leq \ldots \leq p.k_{p.n}$. In p.leaf, the Boolean value is true if p is a leaf and false if p is an internal node. Each internal node p also contains p.n+1 pointers $p.a_1, p.a_2, \ldots, p.a_{p.n+1}$ to its children. Leaf nodes have no children and so their a_i attributes are not defined. The key $p.k_i$ divide the ranges of keys stored in each subtree. If yi is any key stored in the subtree with root $x.a_i$, then $y_1 \leq p.k_1 \leq y_2 \leq p.k_2 \leq \ldots \leq p.k_{p.n} \leq y_{p.n+1}$. All leaves have the same height h. Nodes have lower and upper bounds on the number of keys. The minimum degree of the B-tree is $t \geq 2$. Every node other than the root must have at least t-1 keys. Every internal node other than the root thus has at least t children. If the tree is nonempty, the root must have at least one key. Every node may contain at most 2t-1 keys. Therefore, an internal node may have at most 2t children. A node is full if it contains exactly 2t-1 keys. [18]

6 Proposed Index Structure

The proposed index structure (see in Fig. 4) is constructed by using Hilbert curve and B-tree with inverted file. Before creating the B-tree, two dimensional coordinate points are converted to the single value by using the Hilbert curve. Then, B-tree that combines the inverted file is constructed according to the value from the Hilbert curve (h-values) and services. The inverted file has keywords and location. The Example Dataset is shown in Table 1.

The steps to compute the h-values of Hilbert curve are:

step 1: Input x and y coordinates.
step 2: Convert the x and y coordinates to binary representation.
step 3: Interleave bits of the two binary numbers into one string.
step 4: Separate the string Si from left to right into 2-bit strings (i.e. i=1, …,n).

step 5: Convert to decimal value Di for each 2-bit strings and stores into an array in the same order as the strings occurred. The converting 2-bit string to decimal format is as follow:

1) ‘00’ to 0
2) ‘01’ to 1
3) ‘10’ to 3
4) ‘11’ to 2

step 6: For each i in the array,
If i=0 then change every following occurrence of 1 in the array to 3 and every following occurrence of 3 in the array to 1
If i=3 then change every following occurrence of 0 in the array to 2 and every following occurrence of 2 in the array to 0

step 7: Convert each element in the array to its 2-bit binary strings.

step 8: Concatenate all the strings from left to right and then calculate the decimal value [17].

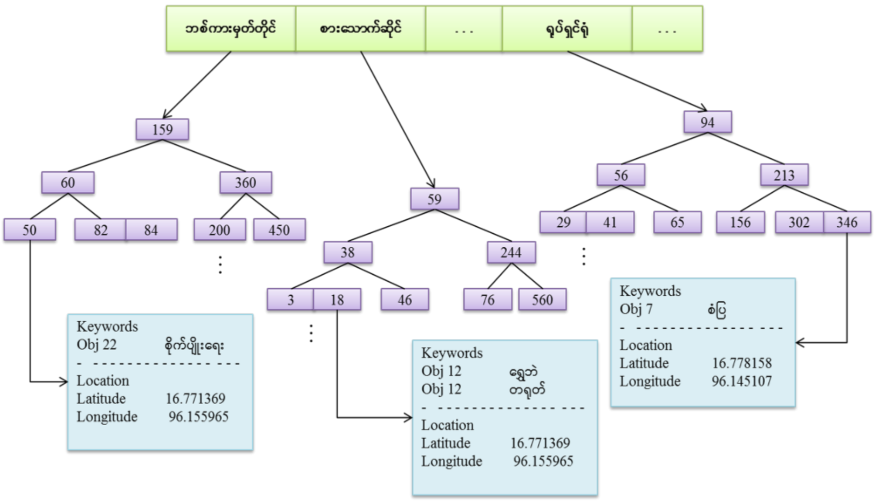

Fig. 4. Proposed Index Structure

Table 1. Example Dataset.

Id	Latitude	Longitude	Keywords	Services
Obj1	16.796433	96.176803	အောင်မင်္ဂလာ	ဘစ်ကား မှတ်တိုင်
Obj2	16.779908	96.140056	ဗဟိုစည်	ဘစ်ကား မှတ်တိုင်
Obj3	16.800442	96.162225	ဗိုလ်ချုပ်	ပန်းခြံ
Obj4	16.829281	96.155644	ဆီဒိုးနား	ဟိုတယ်
Obj5	16.816497	96.127464	ဖူဂျီ, ဂျပန်	စားသောက်ဆိုင်
Obj6	16.810881	96.176419	ရွှေဗဟို	ရုပ်ရှင်ရုံ

7 Experimental Results

In this paper, the locations are searched based on the current location, given range and service. Current location is acquired by GPS and the desired service is chosen by the user. It takes these three inputs and search in the proposed index structure. This system is considered on the mobile devices and is tested on the Yangon Region which has 46 townships. In this system, it is mainly focused on the 20 townships. It provides the user with 72 services and there are 3000 data in the database. Fig. 5 shows Input required query for searching. In Fig 5, user needs to choose the desired service. In Figure 6, it shows the places which are searched by user based on the current location, services and given range.

Figure 7 compares the searching time (second) between using proposed index structure and R-tree and combined K-d tree with inverted file. Searching time using proposed index structure is faster than R-tree and K-d tree with inverted file.

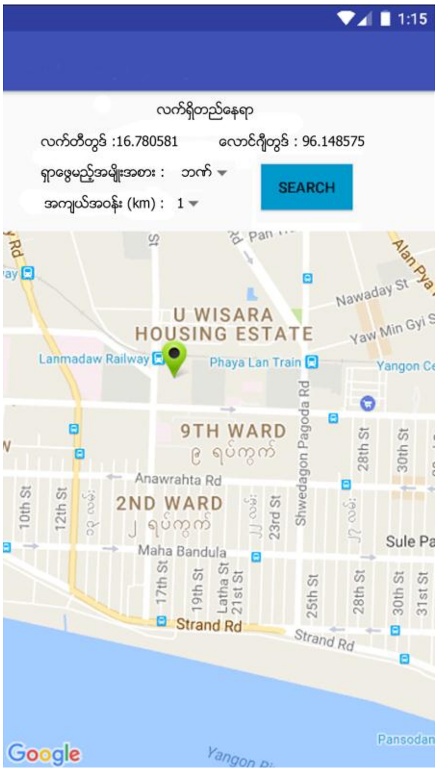

Fig. 5. Input Required Query for Searching

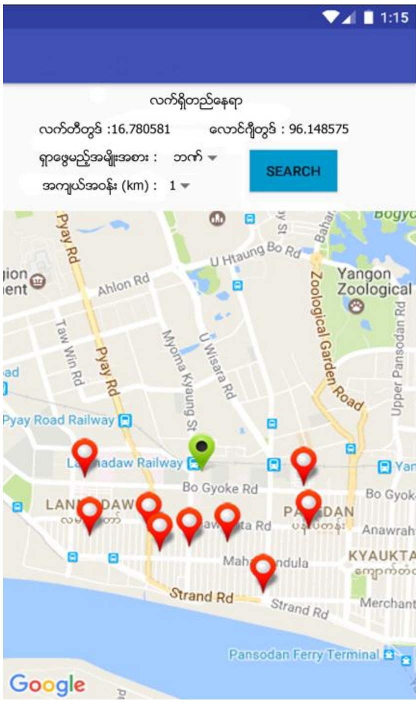

Fig. 6. Result After Searching

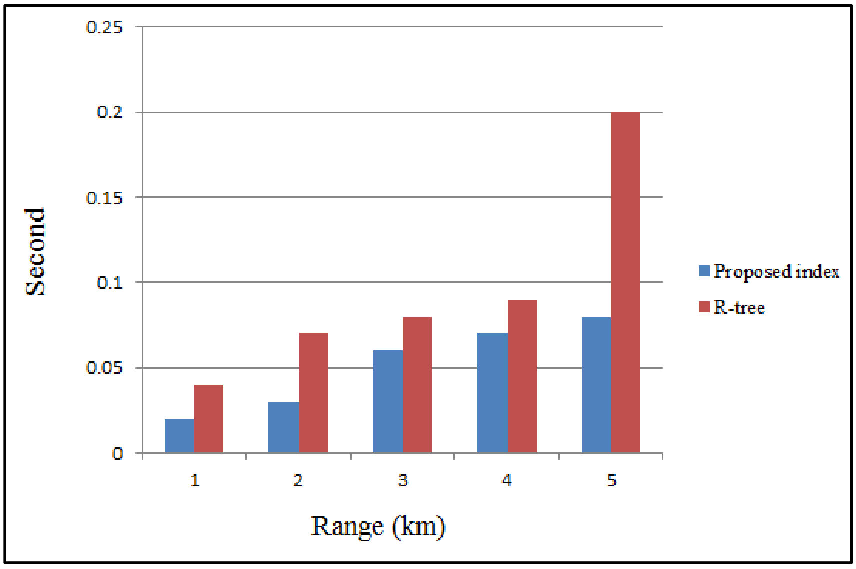

Fig. 7. Searching Time Compare with Proposed Index and R-Tree

8 Conclusions

This paper presents a new index structure that is constructed by using Hilbert space filling curve and B-tree and also combines the inverted file to retrieve the locations with Myanmar language within the given range. This system is intended to develop for the users who are difficult to use with English version. It is tested on Yangon region. This application is considered on the mobile devices. As a further extension, we will consider the system that will search the desired location with both English and Myanmar Language on the mobile devices and will work in an offline.

References

1. S. N. Aung, M. M. Sein, "Hybrid Geo-Textual Index Structure for Spatial Range Keyword Search", Computer Science & Engineering: An International Journal (CSEIJ), Vol. 4, No.5/6, December 2014.
2. S. N. Aung and M. M. Sein, "K-Nearest Neighbours Approximate Keyword Search for Spatial Database", in Proceedings of 9thInternational Conference on Technological Advances in Electrical, Electronics and Computer Engineering (ICTAEECE), Bangkok, Thailand, 7th February 2015, pp. 65-68.
3. S. N. Aung and M. M. Sein, "Index Structure for Nearest Neighbors Search with Required Keywords on Spatial Database", the 9th International Conference on Genetic and Evolutionary Computing (ICGEC 2015), Yangon, Myanmar, 26-28 August 2015.
4. Myat Thiri Khine , Myint Myint Sein, "Geo-Spatial Index Structure for Myanmar Keyword Query", in Proceedings of the 15th International Conference on Computer Applications (ICCA2017), Yangon, Myanmar, February 2017.
5. Myat Thiri Khine, Myint Myint Sein, “Poster: Index Structure for Spatial Keyword Query with Myanmar Language on the Mobile Devices”, in Processing MobiSys '16 Companion Proceedings of the 14th Annual International Conference on Mobile System, Application, and Services Companion, Singapore, pages 43, 2016.
6. X. Cao, L. Chen, G. Cong, C. S. Jensen, Q. Qu, A. Skovsgaard, D. Wu, and M. L. Yiu, “Spatial keyword querying”, inER, 2012, pages 16–29.
7. X. Cao, G. Cong, Christian S. Jensen, Jun.J. Ng, BengC.Ooi, N.T. Phan, D. Wu, “SWROS: A System for the Efficient Retrieval of Relevant Spatial Web Objects”. Available: http://www.ntu.edu.sg/home/gaocong/papers/ vldb12swors.pdf
8. A. Cary, O. Wolfson, and N. Rishe, “Efficient and scalable method for processing top-k spatial Boolean queries”, inSSDBM, 2010, pages 87–95.
9. G. Cong, C. S. Jensen, and D. Wu, “Efficient retrieval of the top-k most relevant spatial web objects”, PVLDB, 2009, 2(1):337–348.
10. I. D. Felipe, V. Hristidis, and N. Rishe, “Keyword search on spatial databases”, in ICDE, 2008, pages 656–665.
11. R. G¨obel, A. Henrich, R. Niemann, and D. Blank, (2009), “A hybrid index structure for geo-textual searches”, in CIKM, 2009, pages 1625–1628.
12. R. Hariharan, B. Hore, C. Li, and S. Mehrotra, (2007), “Processing spatial-keyword (sk) queries in geographic information retrieval (gir) systems”, in SSDBM, 2007, page 16.
13. Z. Li, K. C. K. Lee, B. Zheng, W.-C. Lee, D. L. Lee, and X. Wang, “Ir-tree: An efficient index for geographic document search”, IEEE TKDE, 2011, 23(4):585–599.

14. J. B. Rocha-Junior, O. Gkorgkas, S. Jonassen, and K. Nørv°ag, “Efficient processing of top-k spatial keyword queries”, in SSTD, 2011, pages 205–222.
15. Y. Theodoridis, T. Sellis, “Optimization Issues in R-tree Construction”, Technical Report KDBSLAB-TR-93-08.
16. T. Wang, G. Li, J. Feng, “Efficient Algorithms for Top-k Keyword Queries on Spatial Databases”, 12th IEEE International Conference on Mobile Data Management, 2011.
17. Christos Faloutsos, Shari Roseman, "Fractals for Secondary Key Retrieval", In Proceedings of the Eighth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems ,PODS '89, 247-252.
18. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction To Algroithm", Third Edition, pg 484-499
19. X. Chen, C. Zhang, B. Ge, W. Xiao, " Temporal Social Network: Storage, Indexing and Query Processing", in the Workshop Proceedings of the EDBT/ICDT 2016 Joint Conference, Bordeaux, France, March 15, 2016.
20. Z. Wei , W. Wanzhen, Y. Xingguang, X. Gang, "An optimized query index method based on R-tree," in Proc Fourth International Joint Conference on Computational Sciences and Optimization, pages:1007-1010, 2011.
21. Y. Lifang; L. Rui; H. Xianglin; L. Yueping, "Performance of Rtree with slim-down & Reinsertion Algorithm," in Proc International Conference on Signal Acquisition and Processing, pages:291-294, 2010.
22. Y. Tao and C. Sheng, “Fast Nearest Neighbor Search with Keywords”, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 4, April 2014.
23. Manas Ranjan Kabat, “DESIGN AND ANALYSIS OF ALGORITHMS”, pg 99-105.