COMP214 AI Group Project

**Library Path Searching System (LPSS)**

Final Report

**Team: AI Group2**

**– CodeCooks**

Jiazhe Wang

Wenyang Cai

Xueli Jia

Yiming Li

ZhaoZhao

2012/5/13

**Abstract**

It is always hard and time consuming to find a serious of correct book locations in a big library. To make this more convenience and reduce librarian burden, a *shortest path finding system* is developed. Given a list of wanted books, this system can generated a shortest path which get through all the correct book locations and connect them as circle to indicate the best (shortest) fetching order. This information can be shown on an e-map on UI and user can choice to print it out if (s)he want. As a carrier system, a library management system is also developed for the purpose to simulate the real execution environment of the core system. Like a real library management system, it implement almost all the functionalities that a real library management system should have and additionally a library e-map editor is also implemented. The integration testing result that, the whole integrated system works well in both standard conditions and extreme conditions. On the other hand, in the computing capability of the shortest path finding system, an optimal path can always be found. The result of this report consists of three main parts. Firstly, group member and their corresponding responsibilities are provided. The second part is the details description and analysis of the whole system. At the end of this report, suggestions about system future extension and professional issues are also given.

Contents

[1 People and Roles 4](#_Toc324700177)

[1.1 Team 4](#_Toc324700178)

[1.2 Tasks and Contributions 4](#_Toc324700179)

[1.3 Roles and Responsibilities 5](#_Toc324700180)

[2 Application Overview 5](#_Toc324700181)

[2.1 Application Domain 5](#_Toc324700182)

[2.2 Types of Users 5](#_Toc324700183)

[3 Achievements 5](#_Toc324700184)

[3.1 Webpages 6](#_Toc324700185)

[3.2 Database 6](#_Toc324700186)

[3.3 Algorithms 6](#_Toc324700187)

[3.4 Extra features 6](#_Toc324700188)

[4 Evaluation 6](#_Toc324700189)

[4.1 Algorithm Evaluation 6](#_Toc324700190)

[4.1.1 Algorithm Excellent Performance 6](#_Toc324700191)

[4.1.2 Optimizing Space 7](#_Toc324700192)

[4.2 UI Evaluation 7](#_Toc324700193)

[4.2.1 Usability 7](#_Toc324700194)

[4.2.2 Interface 8](#_Toc324700195)

[4.2.3 Compatibility 8](#_Toc324700196)

[5 Future development 8](#_Toc324700197)

[5.1 Algorithms 8](#_Toc324700198)

[5.2 Functions 8](#_Toc324700199)

[5.3 Compatibility 8](#_Toc324700200)

[5.4 Further consideration 9](#_Toc324700201)

[6 Professional Issues 9](#_Toc324700202)

**Final Report**

# People and Roles

## Team

This project was developed by the team AI Group 2 named CodeCooks. The team members were Jiazhe Wang (Neo), Wenyang Cai (Sidney), Xueli Jia, Yiming Li (Nathan) and Zhao Zhao (sky). All team members were year2 students in Artificial Intelligence of Computer Science at the University of Liverpool.

## Tasks and Contributions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Jiazhe Wang | Wenyang Cai | Xueli Jia | Yiming Li | Zhao Zhao |
| UI Design | Visual Design | Frame Design | - | - | - |
| UI Implement | HTML/CSS /JS develop | - | - | - | - |
| UI Testing | Assist | Testing | - | - | - |
| Algorithm Design | - | - | Assist | - | Design |
| Algorithm Implement | - | - | Java develop | - | Java develop |
| Algorithm testing | - | - | Testing | - | testing |
| Database Design | - | - | - | Design | - |
| Database implement | - | Assist | - | Implement | - |
| Database Testing | - | - | - | Testing | - |
| Website Construction | Architect/ Manage | - | - | Manage/ JSP develop | - |

## Roles and Responsibilities

|  |  |
| --- | --- |
| Jiazhe Wang | Project Architect, Web Engineer (HTML), UI Designer |
| Wenyang Cai | Principal Secretary, Database Administrator |
| Xueli Jia | Algorithm Engineer |
| Yiming Li | Database Designer, Database Engineer, Web Engineer (JSP) |
| Zhao Zhao | Algorithm Designer, Algorithm Engineer |

# Application Overview

## Application Domain

The product was intended to be a system that could find the shortest path to collect particular books in a library. This system was expected to be embedded into a website of library. Libraries have numerous books in a number of bookshelves on different floors. It is quite hard and will cost time to collect a number of books. This product can help a library to construct a user-friendly system for borrowers and improve the efficiency of staffs.

## Types of Users

* **General** **user**:

General users are any LPSS users except employees, may be students or staffs. They can find the locations of books and the shortest path to collect these books.

* **Junior** **Librarian**:

Junior Librarians are people who work for the library, maybe temporary workers. They return books to the right locations and collect books for general user every day. They can use LPSS to improve efficiency or control a library robot to do these jobs.

* **Senior** **librarian**:

Senior librarians are people who manage Junior Librarians and important work of the library. Senior librarian can manipulate the database of books in LPSS. (E.g. change the locations of books on bookshelves)

* **Database** **Administrator**:

Database Administrators are who updates, manipulates and maintains all databases of LPSS. (E.g. update the maps of library)

* **Curator**:

Curators are who has the highest authority. Curator can manage other employees.

# Achievements

The successful accomplishment of LPSS relied on the successful implementations of webpage, database, and algorithms which were three main parts of the system. Almost all functions declared in the last version design document had been implemented well. Most requirements were realized. Moreover, some extra features had been implemented.

## Webpages

Webpages have ability to provide User Interface for users. It helps users to search locations of books and shows them the shortest path to fetch book on the library map. It contains several links accomplishing different functions respectively. For example, the ‘personal’ page stores personal information of users and the ‘search’ page is used to search books and paths.

## Database

Database is used to store data for the system. The database tables designed in the last version design document are all well implemented. For instance, the user table is to store user information including passwords, favorite books. The book table stores all the books provided by a particular library. The map table stores maps of the particular library.

## Algorithms

Algorithms are to implement the functionality of generating the sequence of a list of searching books and then generating a shortest path to fetch these books. Two algorithms, generic algorithm and ant colony algorithm, were implemented to generate the book sequence in Manhattan distance. The results generated by the two algorithms were compared and the better one will be selected as the final result. Finally, the path generate algorithm had the responsibility to generate path on library map according to the generated book sequence.

## Extra features

The additional features spread over all the three parts and each of them will be introduced respectively. About webpage, it implements functions of helping page, downloading user manual, and dealing with the situation of forgetting the password. For database, the storage format has been changed to hardware from main memory for the reason for permanent store data. For algorithm, Path Generator Algorithm has been added to the system to implement function of generating path for fetching searching books in the library map.

# Evaluation

## Algorithm Evaluation

### Algorithm Excellent Performance

From the result of algorithm testing document, the shortest path finding subsystem showed a great execution performance. The ‘great execution performance’ was mainly considered from the algorithm execution speed and computation accuracy point of view.

#### High Accuracy

Both Generic Algorithm and Ant Colony Algorithm have very high computation accuracy for a typical input problem scale. Both of them could generate an optimal solution for the given problem. On the other hand, the shortest path finding subsystem chooses the solution which has less cost. This can guarantee the accuracy of the whole subsystem.

#### High Execution Speed

Both Generic Algorithm and Ant colony Algorithm has very high execution speed for a typical input problem scale. From the testing results shown in algorithm test document: both of these algorithms costs static execution time when the problem scale is less than or equal to twenty-five. From the whole subsystem computation capability point of view, it will take at most four seconds to generate a solution and response to the web server request.

### Optimizing Space

There are two main optimization points for the shortest path finding subsystem.

#### Algorithm Computability

By modifying the variable type manually, the computation of the whole subsystem would extends greatly (extends from 30 books to approximately 40000 books).

#### Path Generator

Algorithm for path generator may need to be redesigned to suit for more complex library map.

## UI Evaluation

### Usability

Most important parts of the system were implemented successfully and worked well in different web browsers. Some assistant functions were implemented to help general users and staffs to use the system.

Staffs could easily manage people via the website. Users could modify their information via the website. Navigations are at the top and left of webpages. Images and fonts structure links can be displayed correctly in multiple major browsers such as chrome (latest), IE9 and Firefox (latest). There was no obvious mistake of content. A help page was provided to help users when they meet some problems of using the website. There were some functions which were not user friendly. Such as the function of adding books to the operation list. Users can only add one book to the operation list every time. There were some functions should be added to the system, such as deleting a book from the library. If a senior librarian adds a wrong book to the library, he would have to delete the book by operating the database. It is inconvenient. Map editing function is not completely implemented. These should be realized in further versions.

### Interface

The website was user friendly and had a nice visual effect. However, some functions had not been implemented yet. These functions should be realized in the further versions.

### Compatibility

The website could be correctly displayed on latest major browser such as chrome, safari, Firefox and IE. All the functions except map editing could be used on those browsers. The UI testing result showed that the website could be correctly displayed in IE9, but not correctly displayed in IE8. What’s more, the map editing function could not be used on Firefox and IE9 because of nonsupport of HTML5. The compatibility of the website was not good for those old versions of browsers. The compatibility could be a problem for this system.

# Future development

## Algorithms

The algorithm was only able to calculate the shortest path in some regular graphics. The shape of the shelves and obstacles should be regular. In future development, the algorithm should be improved to calculate shortest path in all different situation.

## Functions

The function of editing map shall be fully implemented and the staff shall be able to modify library directly on the web. A book deleting function shall be added to the website. A function that allow general user to add multiple books to the operation list shall be added. Some functions that could help staff manage and update library could be added to the website.

## Compatibility

This website cannot be correctly displayed on some old version of the browser and some functions could not be used on some browser. In future development, the compatibility could be improved. The web site could be adapted to more browsers.

## Further consideration

As each module of the system was relatively independent, many useful components can be transplanted to other systems. For example, the heuristic path finding algorithm can be transplanted to other systems involved in path searching. This can improve reuse of applications. The library management system is only a carrier of the core system. That means this system can be transferred or even embedded in various kinds of carrier system. For example, this path searching system can also be embedded in an automatic forklift in factory warehouse. It even can be applied in mobile aid software to help the post man find the shortest path. In addition, if the library has a library robot, the system could be modified to add the function to control the robot.

# Professional Issues

The British Computer Society (BCS) enables individuals, organizations and society to realize the potential of and maximize the benefits from IT by setting and maintaining the highest professional standards for IT professionals. For this library system, conducts were applied and complied with as follows:

1. The team had studied and been familiar with the database and structure of library.
2. This website and system did not involve any problem with public health, safety and environment.
3. This product would not affect third parties. The product was original and has nothing copied from other companies.
4. The database had been encrypted and the password would also be encrypted by Message-Digest Algorithm 5(MD5). Users’ personal information was kept safely.
5. The language of website was English and all users have the same right to access the website to search books.
6. No bribery or inducement happened in this project.
7. All data and webpages were created by the team without using codes from other projects, conflict of interest was avoided. References were listed in the document.
8. The team did not disclose or authorize to be disclosed, or use for personal gain or to benefit a third party, or at the direction of a court of law.
9. User help functions were provided to let users know how to search books. They also represent the right information to staffs and general users.
10. The team sought to improve professional standards through participation in development, use and enforcement, such as adding some comments and making the website user-friendly.
11. The team acted with integrity in relationships with all members of the BCS.
12. The team had due regard for the possible consequences of the statements on others.
13. The team was nonprofit. Problems like bankrupt would not be involved.
14. We had used JSP to develop our website. Some new technologies had been used in the project, such as HTML5.
15. Some members of the team had the experience of designing database and using JSP, so the team had the competence to complete this project.
16. The team had observed the relevant BCS Codes of Practice and convention of codes.
17. The team had meetings more than twice a month. Tasks were completed well and on time.

# Bibliography

[1] Berge, Claude (1958) (in French). Théorie des graphes et ses applications. Dunod, Paris: Collection Universitaire de Mathématiques, II. pp. viii+277. Translation:. Dover, New York: Wiley. 2001 [1962].

[2] Wikipedia, available at: <http://en.wikipedia.org/wiki/Undirected_graph#Undirected_graph>

[3] “Der Handlungsreisende – wie eri sein soll und was er zu thun [sic] hat, um Aufträge zu erhalten und eines glücklichen Erfolgs in seinen Geschäften gewiß zu sein – von einem alten Commis-Voyageur” (The traveling salesman — how he must be and what he should do in order to be sure to perform his tasks and have success in his business — by a high commis-voyageur)

[4] A detailed treatment of the connection between Menger and Whitney as well as the growth in the study of TSP can be found in Alexander Schrijver's 2005 paper "On the history of combinatorial optimization (till 1960). Handbook of Discrete Optimization (K. Aardal, G.L. Nemhauserr, R. Weismantel, eds.), Elsevier, Amsterdam, 2005, pp. 1–68.

[5] Wikipedia, available at: <http://en.wikipedia.org/wiki/NP-hard>

[6] carpaneto G，Toth P．Some New Branching and Bounding Criteria for the Asymmetric Traveling Salesman Problem [J]．Management Science，1980，(26)

[7] Dantzig G B．Solution of a Large Scale Traveling Salesman Problem [J]．Operations Research，1954，(2)．

[8] Bellman R．Dynamic Programming Treatment of the Traveling Salesman Problem [J]，J．ACM，1962，(9)．

[9] Grefenstette J J，Gopal R Rosmaita B，et a1．Genetic Algorithms for Traveling Salesman Problem[A[．In：Proceedings of an International Conference on Genetic Algorithms and Their Applications[C]，1985．

[10] Cheng—Fa Tsai，Chun—Wei Tsai，Ching-Chang Tseng．A new hybrid heuristic approach for solving large traveling salesman problem [J]．Information Sciences，2004，(166)

[11] Gregory Gutin，Andem Yeo． Polynomial approximation algorithms for the TSP and the QAP with a factorial domination number[J]．Discrete Applied Mathematics，2002。(1 19)．

[12] E．M．Cochranej．E．Beasley．111e CO-adaptive neural network approach to the Euclidean Traveling Salesman Problem [J]．Neural Networks，2003。(16)．

[13] V．Deineko．New exponential neighborhood for pelynomially solvable TSPs·[J]．Electronic Notes in Discrete Mathematics．2004，(17)．

[14] Hans-Joachim B~kenhauer，Juraj Hromkovi．C， RalfKlasing，et a1．Towards the notion of stability of approximation for hard optimization tasks and the traveling salesman problem[J]．Theoretical Computer Science，2002，(285)．

[15] J&6me Monnot．Diferential approximation results for the traveling salesman and related problems[J]． Information Processing Letters，2002，(82)．

[16] Forbes J．Burkowski．Proximity and priori~：applying a gene expression algorithm to the Traveling Salesperson ProblemU]．Parallel Computing，2004，(30)．

[17] Shang Gao. Chaotic Ant Colony Algorithm for traveling sales problem [J]. Systems Engineering Theory and Practice, 2005. (9)

[18] Xiehua Sun. simulated annealing algorithm for traveling sales problem [J]. Journal of China Metrology institute, 2005. (1)

[19] ^ [a](http://en.wikipedia.org/wiki/Natural_selection#cite_ref-origin_0-0) [b](http://en.wikipedia.org/wiki/Natural_selection#cite_ref-origin_0-1) Darwin C (1859) [On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life](http://en.wikipedia.org/wiki/On_the_Origin_of_Species_by_Means_of_Natural_Selection,_or_the_Preservation_of_Favoured_Races_in_the_Struggle_for_Life) John Murray, London; modern reprint Charles Darwin, Julian Huxley (2003). On The Origin of Species. Signet Classics. [ISBN](http://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-451-52906-5](http://en.wikipedia.org/wiki/Special:BookSources/0-451-52906-5). Published online at [the complete work of Charles Darwin online](http://darwin-online.org.uk/): [On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life](http://darwin-online.org.uk/content/frameset?itemID=F373&viewtype=side&pageseq=2)

[20] Thomas M. Connolly, Carolyn E. Begg, *Database Solutions*, Addison Wesley, 2004.

[21]Nik(2010). *Subway Map Jquery Plugin*. Available from http://www.kalyani.com/2010/10/subway-map-visualization-jquery-plugin/ (13 May 2012)

[22] Thomas M. Connolly, Carolyn E. Begg, Database Solutions, Addison Wesley, 2004.

[23] Ian Sommerville, Software Engineering, Assison-Wesley, 2006, p491.

[24] Lientz, B.P. and Swanson, E.B., Software Maintenance Management, A Study Of The Maintenance OfComputer Application Software In 487 Data Processing Organizations. Addison-Wesley, Reading MA, 1980. ISBN 0201042053