[[1]](#footnote-1)

Encode and Decode Morse

*Name students*

*Phan Sỹ Nhật Tân – 20192247- ĐKTĐH- HTĐ Name Instructor*

*Nguyễn Hoàng Anh– 20192183 - ĐKTĐH- HTĐ TS. Hoàng Đức Chinh*

*Nguyễn Thành Hưng– 20192211- ĐKTĐH- HTĐ*

***Abstract*— Morse code is a method often used together with an electrical telegraph to transfer text message remotely in late 19th century until early 20th century and it is a method used in telecommunication to encode text characters as standardized sequences of two different signal durations and Morse code is named after Samuel Morse, one of the inventors of the telegraph. People also send message encoded with Morse code by flashing light or waving flag. The Morse code encodes text characters as sequences of two different signal durations, call dots and dashes, or dits and dahs, i.e., “.” And “-”. The dots represent short signal, and the dashes represent the long one.**

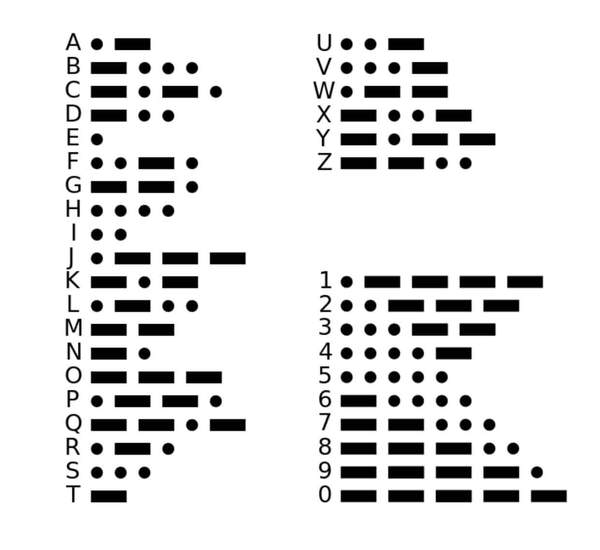
***Keyworks*— Morse code; inventors; encoded code; represent.**

# I. Induction

M

orse code is a type of character encoding used in telecommunications to transmit telegraph information. Morse code uses a normalized sequence of long and short elements to represent letters, numbers, periods, dashes, and special characters of messages. Morse code is still used by amateur broadcasters today, and it is also very useful in transmitting emergency signals in an emergency.

In this report, we show a program in C++ that opens a file and finds if it contains text data (English) or Morse code. It will then convert the file to another format, i.e., if the file contains text data, the program will convert the normal text into Morse code and save the result in another file. The program can handle any error that may occur by displaying a message with an error code, from which the user can identify the error code and correct it.



*Morse code*

# II. The idea, design, and implementation of the program

* The idea

As following the project, the requirement for our team is that: we need to find a programming method which we can code simultaneously and finely delineating work. Besides, we need reusable function, inheritance of a function to other functions to compacting and simplyfying. Finally, in order to optimize our project, we need a good algorithm with most suitable time complexity which can deal with a large amount of data from file. So, we came up with ideas and used mainly these methods:

* **Three – layer architecture**
* **Data transfer object class**
* **Object- Oriented Programming (OOP)**
* **Hash table**
* **Linked list**

## A. Three – layer architecture

1. *What’s three-layer architecture?*

Layer indicates the logical separation of components. Layered architecture concentrates on grouping related functionality within an application into distinct layers that are stacked vertically on top of each other. Each layer has unique namespaces and classes.



*Three – layer architecture*

1. *Information about three-layer follows:*
2. **Presentation layer**

It is the first and topmost layer present in the application where users can interact with the application.

1. **Business Logic layer**

This is the middle layer - the heart of the application. It contains all business logic of the application, describes how business objects interact with each other, where the Presentation layer and Data Access layer can indirectly communicate with each other.

1. **Data Access layer**

The Data Access layer where Data is stored in Database and get data from Database.

1. *Advantages of three-layer architecture*

• Explicit code: The code is separated into each layer. Each one is dedicated to a single responsibility such as interface, business processing, and querying instead of bundling all the code in one place.

• Easy to maintain: As its roles separate each layer, it would be easier to change something. The modification can be isolated in a single layer or affected

only the nearest layer without affecting the whole program.

• Easy to develop, reuse: When we want to modify a function, we can easily do that as we already have a standard architecture. In case we want to alter a complete layer such as from Winform to Web form, we only need to implement to replace the Presentation layer; other layers can be reused completely.

• Easy to transfer: We could save time on moving the application to others as they have a standard architecture to follow and apply.

• Easy to distribute the workloads: By organizing the code into different layers based on its responsibility, each team/member can write their code on each layer independently, which in turn helps developers control their workload.

1. *How does three-layer architecture work?*

In three-layer architecture, the Presentation layer doesn’t communicate directly with the Data Access layer. The Business Logic layer works as a bridge between the Presentation layer and the Data Access layer. The three-layer architecture works as follows:

• The Presentation layer is the only class that is directly interacted with the user. It is mainly used for presenting/collecting data from users, then passes them to the Business Logic layer for further processing.

• After receiving information from the Presentation layer, the Business Logic layer does some business logic on the data. During this process, this layer may retrieve or update some data from the application database. However, this layer doesn’t take responsibility for accessing the database; it sends requests to the next layer, the Data Access layer.

• The Data Access layer receives requests from the Business Logic layer and builds some queries to the database to handle these requests. Once the execution gets done, it sends the result back to the Business Logic layer.

• The Business Logic layer gets responses from the Data Access layer, then completes the process and sends the result to the Presentation Layer.

• The Presentation layer gets the responses and presents them to users via UI components.

* *An addition layer: Dto*

1. What is the Data transfer object class?

DTOs or Data Transfer Objects are objects that carry data between layers Gui, Dal, Bll in order to reduce the number of methods calls.

1. Dto layer structure in our project including

Three lists: Text list, Morse list, Error list

Fields: CompletedTime, CompletedDate, CompletedTimeDate, Duration, TotalWord, SuccessedWord, ErrorWord, TotalCharacter, ConvertedCharacter, ErrorCharacter.

## B. Object- Oriented Programming (OOP)

1. *What is the OOP?*

Object-Oriented Programming (or OOP) is a paradigm of programming in which programs are written and structured around objects rather than functions or logic. Here, objects are defined as data fields that have unique attributes and behavior. They contain data in the form of attributes and procedures in the form of methods. Object

procedures can access and modify the data present in an object.

1. *Why we choose OOP for our project?*

We can build the programs from standard working modules that communicate with one another, rather than having to start writing the code from scratch which leads to saving of development time and higher productivity,

OOP language allows to break the program into the bit-sized problems that can be solved easily (one object at a time). OOP systems can be easily upgraded from small to large systems. It is possible that multiple instances of objects co-exist without any interference. It is very easy to partition the work in a project based on objects.

It is possible to map the objects in problem domain to those in the program. By using inheritance, we can eliminate redundant code and extend the use of existing classes. Message passing techniques is used for communication between objects which makes the interface descriptions with external systems much simpler. The data-centered design approach enables us to capture more details of model in an implementable form.

## C. Data structure

* *Hash table*

1. *What is the hash table?*

In computing, a hash table (hash map) is a data structure that implements an associative array abstract data type, a structure that can map keys to values. A hash table uses a hash function to compute an index.

If all keys are known ahead of time, a perfect hash function can be used to create a perfect hash table that has no collisions. If minimal perfect hashing is used, every location in the hash table can be used as well. We can access a key with time complexity O(1).

1. *Applying to our project*

As the of result of knowing all the morse codes of all characters, we can build a perfect hash table. In addition, in order to encode texts to morse codes and interpreter morse codes to texts, we try to build two hash tables for each situation.

Diagram

Description automatically generated

1. *Hash table for Interpreting*

* First of all, writing a hash function and trying to find a perfect hashing. In order to make sure no collision, we write a c++ program to evaluate that. In our project, we transfer a morse code to hash value (an integer number) by using a hash function. In this function, if the symbol is a “.” (dot), the hash value will be increased by the Ith (index) power of 2. And if the symbol is “-” (dash), hash value will be increased (I+1) the power of 2.
* Second, we save hash value with the character respectively in a text file with file name is “texthash.txt”.

**Graphical user interface, application

Description automatically generated**

*Format of texthash.txt*

1. *Hash table for Encoding*

* First, as same as with interpreting, writing a hash function and trying to find a perfect hashing. In our project, we transfer a morse code to hash value (an integer number) by using a hash function. In this function, hash value is equal the integer value of each character in Ascii table. Because of Ascii value is distinct, so the hash table is perfect.
* Second, we save hash value with the morse code respectively in a text file with file name is “morsehash.txt”.

Graphical user interface, application

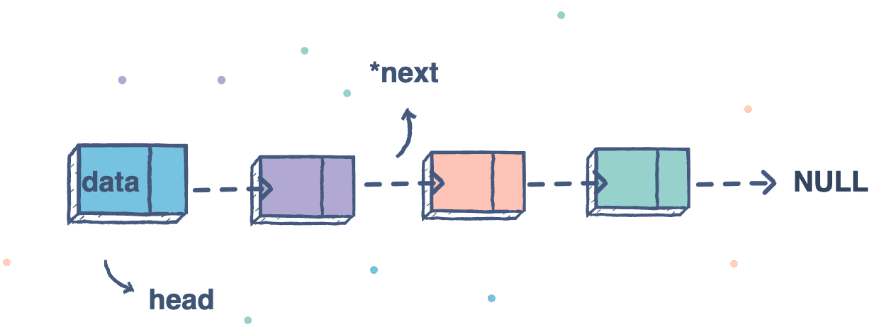
Description automatically generated

*Format of morsehash.txt*

* *Linked list*

1. *What is the linked list?*

In computer science, a linked list is a linear collection of data elements whose order is not given by their physical placement in memory. Instead, each element points to the next. It is a data structure consisting of a collection of nodes which together represent a sequence. In its most basic form, each node contains: data, and a reference (in other words, a link) to the next node in the sequence. This structure allows for efficient insertion or removal of elements from any position in the sequence during iteration. More complex variants add additional links, allowing more efficient insertion or removal of nodes at arbitrary positions.



*Singly linked list*

1. *Applying singly linked list to project*

Singly linked list is suitable for our project because we have not to assign a fixed array to deal with getting data. The list can be increased its size automatically. It is suitable for store data type such as Error, Morse Code and Text which we do not know the amount of

We use standard library: list to get the data structure list instead of writing ourselves linked list, because it is more optimize.

## D. The algorithm

Applying the idea of using a hash table, the whole hash will be uploaded from Dal, then the morse or text string from the input file will be cut and then passed through the hash function to convert it to text or morse respectively. All errors will be saved during this phase.

*Two main functions are presented by followchart (on final page).*

# III. Discussion and evaluation of the results

* *A few optional commands can be:*

o –h: print out the help to instruct user of the usage on the screen and save the instruction to a file name readme.txt, override if readme.txt has already existed.

Text

Description automatically generated

o –t: force to take the input file as a text file and the output file as the Morse code one.

Text

Description automatically generated

o –m: force to take the input file as Morse code and the output file as the interpreted text file.



o –c: print the statistics of the successful conversion to the screen.

Text

Description automatically generated

* *The program can handle any error that may occur by displaying a message with an error code from which the student can correct the lesson. Some cases are listed below.*

If output file has been existed, the output: Warning: D:\\output.txt already exists. Do you wish to overwrite (y, n)? (1)

Text

Description automatically generated(1)

The program cannot open the input file. Message can be: “Error XX: FILENAME could not be opened”(2)

A screenshot of a computer

Description automatically generated with low confidence(2)

If the text file contains characters which do not have the corresponding ones in Morse code chart (3)

Text

Description automatically generated(3)

If the Morse code file contains the wrong format codes(e.g. total number of “.” and “-” is larger than 7), each wrong code should be converted to # in text. If Morse code is in the right format but unrecognized, i.e. it does not have the corresponding letter character, it should be converted to “\*” in the output file. In both the cases, the error message like: “Error AB: Invalid Morse code CODE on line XX.” should be shown where “CODE” is the Morse code that cannot be converted.(4)(5)

Graphical user interface, text

Description automatically generated(4)

Text

Description automatically generated

(5)

If user enter an unrecognized command, the program should exit and print out an error message “Error XX: Unknown command. Type “morse –h” for help” (6)

Text, logo

Description automatically generated(6)

* *What have been done and have not been done in our work:*

**We have completed all the cases required by the assignment.**

**In the hash table convert from morse code to text we have not tested 100% of the cases to avoid collisions.**

# IV. Work assigned

|  |  |  |
| --- | --- | --- |
| Task | Description of work | Implementer |
| 1. Finding idea |  | All |
| 2. Design algorithm flowchart | From the idea, sketching the program and designing algorithm flowchart. | Nguyen Hoang Anh (Leader) |
| 3.  Presentation Layer (GUI) | Design based on the requirements of the topic and get data from BLL for processing. | Phan Sy Nhat Tan |
| 4. Business Logic Layer (BLL) | Based on flowchart to write algorithm, get and process data from GUI layer, pass data to DAL layer. | Nguyen Hoang Anh (Leader) |
| 5. Data Access Layer (DAL) | Receive information from the BLL layer read the corresponding information and process the data. | Nguyen Thanh Hung |
| 6. Fixing bugs |  | Nguyen Thanh Hung /Phan Sy Nhat Tan |
| 7. Report | Write a report using a sample from IEEE to explain. | All |

References

1. GeeksforGreeks, “List in C++ Standard Template Library (STL)”,5th Floor, A-118, Sector-136, Noida, Uttar Pradesh- 201305. [Online]. Available: https://www.geeksforgeeks.org/list-cpp-stl/
2. TOPDev, “Mô hình ba lớp (Three-layer)”, Viet Nam. [Online]. Available: <https://topdev.vn/blog/mo-hinh-3-lop-la-gi/>
3. Wikipedia,” Hash table”, [Online]. Available: <https://en.wikipedia.org/wiki/Hash_table>
4. LabVIETCHEM, Viet Nam,” Mật mã Morse”. [Online]. Available: https://labvietchem.com.vn/tin-tuc/mat-ma-morse-la-gi.html

Diagram

Description automatically generated

*Morse to text*

Diagram

Description automatically generated

*Text to morse*

1. [↑](#footnote-ref-1)