



Project 2

Morse Code Decoder/Encoder

<https://github.com/Nate314/MorseCode>

Nathan Gawith

Assumptions

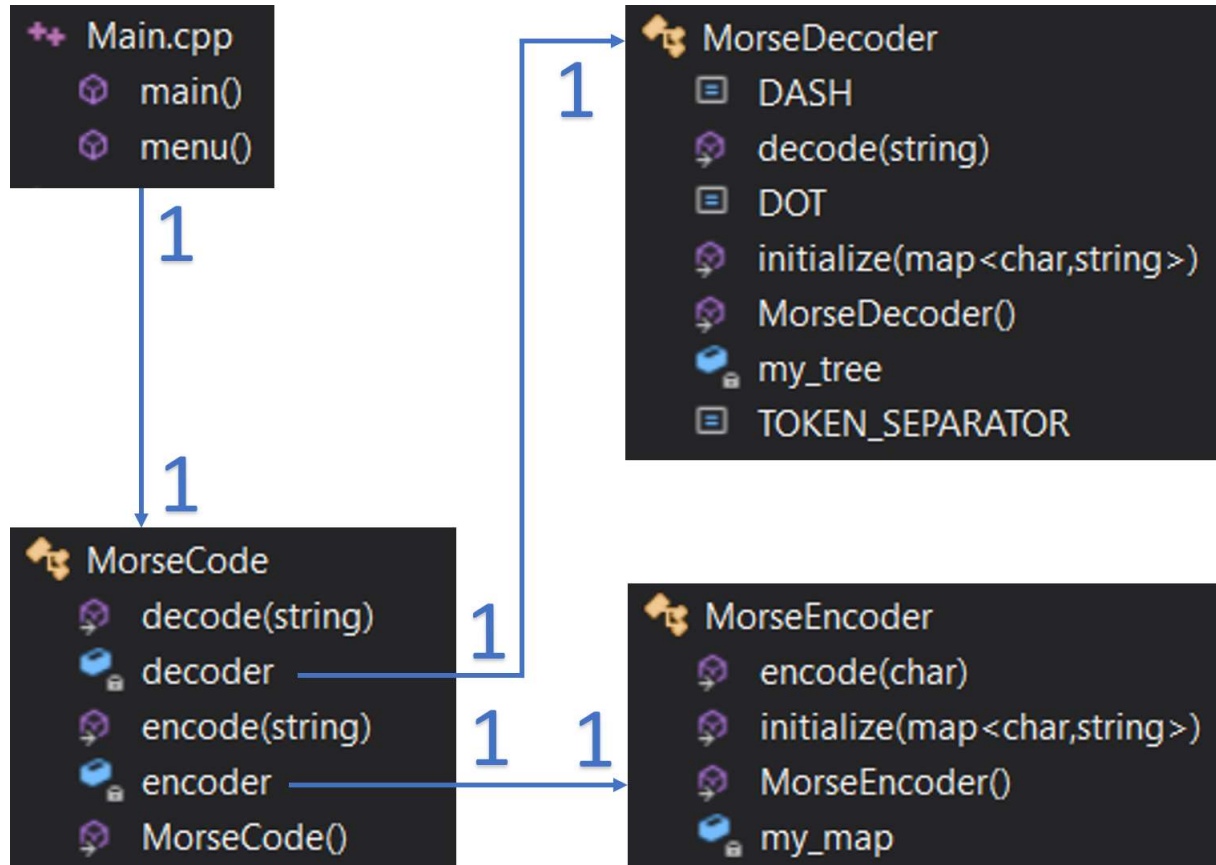
This program does not enter any spaces. If you enter more than one word, the spaces will be removed.

I used the Binary_Tree and BTNode classes from the lecture.

The key value pairs are loaded in from dictionary.txt, which has the same exact file contents as morse.txt at this link <https://www.dropbox.com/s/3cj8yb8gcdsrefg/morse.txt?dl=0>

Input will always be only lowercase letters

UML Diagram



Main

- `main(): int`
- `menu(): void`
- **MorseCode**
 - `encode(string): string`
 - `decode(string): string`
 - **MorseDecoder**
 - `initialize(map<char, string>): void`
 - `decode(string): char`
 - **MorseEncoder**
 - `initialize(map<char, string>): void`
 - `encode(char): string`

Big-O of Algorithms

void MorseDecoder::initialize(map):

```
...  
For  
    ...  
    insert()
```

void MorseEncoder::initialize(map):

```
For  
    ...
```

MorseCode():

```
...  
While  
    ...  
encoder.initialize(morse_codes)  
decoder.initialize(morse_codes)
```

char MorseDecoder::decode(string):

```
...  
For  
    ...
```

string MorseEncoder::encode(char):

```
...
```

string MorseCode::encode(string):

```
...  
While  
    encoder.encode(char)
```

string MorseCode::decode(string):

```
...  
While  
    decoder.decode(string)
```

The insert() function is $\lg n$

$F(n) = c + (n * F(\text{insert}()))$

$F(n) = c + (n * \lg n)$

$O(n) = n \lg n$

This initialize function is just copying over the map

$F(n) = n$

$O(n) = n$

The constructor does n statements and then calls the above two initialize functions

$F(n) = c + n + F(e.\text{init}()) + F(d.\text{init}())$

$F(n) = c + n + (n \lg n) + (n)$

$F(n) = c + 2n + n \lg n$

$O(n) = n$

Decode traverses the tree until it finds the char

$F(n) = \lg n$

$O(n) = \lg n$

encode() simply looks up a value in a map

$F(n) = c + 1$

$O(n) = 1$

Calls MorseEncoder::encode() in a loop

$F(n) = c + (n * F(e.\text{encode}()))$

$F(n) = c + (n * (1))$

$O(n) = n$

Calls MorseDecoder::decode() in a loop

$F(n) = c + (n * F(d.\text{decode}()))$

$F(n) = c + (n * (\lg n))$

$O(n) = n \lg n$

References: Cite any references that you used.

Morse Code Converter used for testing -

<https://morsecode.scphillips.com/translator.html>

General Help Links

<https://stackoverflow.com/questions/110157/how-to-retrieve-all-keys-or-values-from-a-stdmap-and-put-them-into-a-vector>

<https://stackoverflow.com/questions/743203/accessing-static-class-variables-in-c>

<http://www.cplusplus.com/reference/sstream/stringstream/str/>

<https://www.geeksforgeeks.org/pair-in-cpp-stl/>

How to evaluate postfix expressions

Canvas > CS303 > Modules > Source Code > Binary Trees, Binary Search Trees