Encryption_Decryption

Generated by Doxygen 1.8.14

Contents

Chapter 1

Encryption & Decryption program

1.1 program encrypt & decrypt the given messages

Chapter 2

Encryption_Decryption

Chapter 3

Bug List

File main.c

No bugs

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Chapter 5

File Documentation

5.1 decryption.c File Reference

```
#include <stdio.h>
#include <string.h>
#include "encr_decr.h"
```

Functions

- void decryption (char *message)

  This function decrypt the message given by user.

5.1.1 Function Documentation

5.1.1.1 decryption()

```
void decryption (
            char * message )
```

This function decrypt the message given by user.

Author

Nikola Petrovic

Parameters

*message	We are getting the message added by user from this parameter

This represent an empty string where we are going to put our decrypted message.

Number of characters in a row. Row is given by number next to 'n'.

Arrays of characters which are located in a given row.

Getting a length of message.

Shows the position of letter 'a' in decrypted message.

Pointers to given arrays and strings.

Letter for decryption is a first letter of a given message.

Calculate the distance between 'a' and letter for decryption.

Starting from the first character of message.

Decryption with first row

Starting from the first character of the first row.

Creating a circular moving for first row

If position of a character is smaller than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert decrypted character.

If position of a character is not smaller than the number of characters in a row, inserting of decrypted character is proceed like this.

Decryption with second row

Starting from the first character of the second row.

Creating a circular moving for second row

If position of a character is smaller than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert decrypted character.

If position of a character is not smaller than the number of characters in a row, inserting of decrypted character is proceed like this.

Decryption with third row

Starting from the first character of the third row.

Creating a circular moving for third row

If position of a character is smaller than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert decrypted character.

If position of a character is not smaller than the number of characters in a row, inserting of decrypted character is proceed like this.

If character is not found in any of given rows, just insert a character from a given message.

5.2 encr_decr.h File Reference

Functions

- void encryption (char *message)

  Functions prototypes are declared in this header file.
- void decryption (char *message)

  This function decrypt the message given by user.

5.2.1 Function Documentation

5.2.1.1 decryption()

```
void decryption (
            char * message )
```

This function decrypt the message given by user.

Author

Nikola Petrovic

Parameters

*message	We are getting the message added by user from this parameter

This represent an empty string where we are going to put our decrypted message.

Number of characters in a row. Row is given by number next to 'n'.

Arrays of characters which are located in a given row.

Getting a length of message.

Shows the position of letter 'a' in decrypted message.

Pointers to given arrays and strings.

Letter for decryption is a first letter of a given message.

Calculate the distance between 'a' and letter for decryption.

Starting from the first character of message.

Decryption with first row

Starting from the first character of the first row.

Creating a circular moving for first row

If position of a character is smaller than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert decrypted character.

If position of a character is not smaller than the number of characters in a row, inserting of decrypted character is proceed like this.

Decryption with second row

Starting from the first character of the second row.

Creating a circular moving for second row

If position of a character is smaller than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert decrypted character.

If position of a character is not smaller than the number of characters in a row, inserting of decrypted character is proceed like this.

Decryption with third row

Starting from the first character of the third row.

Creating a circular moving for third row

If position of a character is smaller than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert decrypted character.

If position of a character is not smaller than the number of characters in a row, inserting of decrypted character is proceed like this.

If character is not found in any of given rows, just insert a character from a given message.

5.2.1.2 encryption()

```
void encryption (
            char * message )
```

Functions prototypes are declared in this header file.

Author

Nikola Petrovic

Parameters

∗message	We are getting the message added by user from this parameter

Functions prototypes are declared in this header file.

Author

Nikola Petrovic

Parameters

∗message	We are getting the message added by user from this parameter

This represent an empty string where we are going to put our encrypted message.

Number of characters in a row. Row is given by number next to 'n'.

Arrays of characters which are located in a given row.

Getting a length of message.

Shows the position of letter 'a' in encrypted message.

Pointers to given arrays and strings.

Calculate the distance between 'a' and letter for encryption that is introduced by user.

Starting from the first character of message.

Encryption with first row

Starting from the first character of the first row.

Creating a circular moving for first row

If position of a character is greater than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert encrypted character.

If position of a character is not greater than the number of characters in a row, inserting of encrypted character is proceed like this.

Encryption with second row

Starting from the first character of the second row.

Creating a circular moving for second row

If position of a character is greater than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert encrypted character.

If position of a character is not greater than the number of characters in a row, inserting of encrypted character is proceed like this.

Encryption with third row

Starting from the first character of the third row.

Creating a circular moving for third row

If position of a character is greater than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert encrypted character.

If position of a character is not greater than the number of characters in a row, inserting of encrypted character is proceed like this.

If character is not found in any of given rows, just insert a character from a given message.

5.3 encryption.c File Reference

```
#include <stdio.h>
#include <string.h>
#include "encr_decr.h"
```

Functions

- void encryption (char *message)

  This function encrypt the message given by user.

5.3.1 Function Documentation

5.3.1.1 encryption()

```
void encryption (
            char * message )
```

This function encrypt the message given by user.

Functions prototypes are declared in this header file.

Author

Nikola Petrovic

Parameters

*message	We are getting the message added by user from this parameter

This represent an empty string where we are going to put our encrypted message.

Number of characters in a row. Row is given by number next to 'n'.

Arrays of characters which are located in a given row.

Getting a length of message.

Shows the position of letter 'a' in encrypted message.

Pointers to given arrays and strings.

Calculate the distance between 'a' and letter for encryption that is introduced by user.

Starting from the first character of message.

Encryption with first row

Starting from the first character of the first row.

Creating a circular moving for first row

If position of a character is greater than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert encrypted character.

If position of a character is not greater than the number of characters in a row, inserting of encrypted character is proceed like this.

Encryption with second row

Starting from the first character of the second row.

Creating a circular moving for second row

If position of a character is greater than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert encrypted character.

If position of a character is not greater than the number of characters in a row, inserting of encrypted character is proceed like this.

Encryption with third row

Starting from the first character of the third row.

Creating a circular moving for third row

If position of a character is greater than the number of characters in a row, position of a that character is calculated with this formula.

In position of character of a given message insert encrypted character.

If position of a character is not greater than the number of characters in a row, inserting of encrypted character is proceed like this.

If character is not found in any of given rows, just insert a character from a given message.

5.4 main.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "encr_decr.h"
```

Functions

- int main ()

5.4.1 Detailed Description

Author

Nikola Petrovic

Bug No bugs

5.4.2 Function Documentation

5.4.2.1 main()

```
int main ( )
```

In this variable we are storing the messages added by user.

Variable for choice.

The problem was that the newline I input when reading choice was being taken as input for the next gets(message), and so it was reading an empty string. Because of that I added an artificial getchar after scanf to absorb the newline.

In this case we encrypt the message given by user.

In this case we decrypt the message given by user.

5.5 README.md File Reference

Index