Encryption_Decryption

Generated by Doxygen 1.8.14

Contents

Index

1	Enci	ryption	& Decryp	tion p	rogram	1													1
	1.1	progra	m encrypt	& deci	rypt the	e give	en m	ess	ages			 	 	 			 		1
2	Enci	ryption	_Decrypti	on															3
3	Bug	List																	5
4	File	Index																	7
	4.1	File Lis	st									 	 	 			 		7
5	File	Docum	entation																9
	5.1	decryp	otion.c File	Refere	ence .							 	 	 			 		9
		5.1.1	Function	Docur	nentati	on .						 	 	 			 		9
			5.1.1.1	decry	yption()							 	 	 			 		9
	5.2	encr_c	decr.h File	Refere	ence .							 	 	 			 		11
		5.2.1	Function	Docur	nentati	on .						 	 	 			 		11
			5.2.1.1	decry	yption()							 	 	 			 		11
			5.2.1.2	encry	yption()							 	 	 			 		12
	5.3	encryp	otion.c File	Refere	ence .					٠.		 	 	 			 		14
		5.3.1	Function	Docur	mentati	on .						 	 	 			 		14
			5.3.1.1	encry	yption()			٠.				 	 	 			 		14
	5.4	main.c	File Refe	rence				٠.				 	 	 			 		15
		5.4.1	Detailed	Descri	iption							 	 	 			 		16
		5.4.2	Function	Docur	nentati	on .						 	 ٠.	 			 		16
			5.4.2.1		()														16
	5.5	READ	ME.md Fil	e Refe	rence						٠.	 	 	 ٠.			 		16

17

Encryption & Decryption program

1.1 program encrypt & decrypt the given messages

Encryption_Decryption

Bug List

File main.c No bugs 6 Bug List

File Index

4.1 File List

Here is a list of all files with brief descriptions:

decryption.c				 													 					9
encr_decr.h				 													 					11
encryption.c				 													 					14
main.c				 													 					15

8 File Index

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()

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()

This function decrypt the message given by user.

Author

Nikola Petrovic

Parameters

*message	We are getting the message added by user from this parameter
Millobbago	1 110 are getting the incodage added by abor norm tine 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()

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 a	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()

This function encrypt the message given by user.

Functions prototypes are declared in this header file.

Author

Nikola Petrovic

Parameters

* <i>m</i>	essage	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.

5.4 main.c File Reference 15

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

```
decryption
    decryption.c, 9
    encr_decr.h, 11
decryption.c, 9
    decryption, 9
encr_decr.h, 11
    decryption, 11
    encryption, 12
encryption
    encr_decr.h, 12
    encryption.c, 14
encryption.c, 14
    encryption, 14
main
    main.c, 16
main.c, 15
    main, 16
README.md, 16
```