

# プログラミング演習I 課題ドキュメント

b162392

構築: Doxygen 1.8.6

2016 年 12 月 15 日 (木) 14 時 28 分 32 秒

## Contents

1	ファイル詳解	1
1.1	main.c ファイル	1
1.1.1	詳解	1
1.1.2	関数詳解	1
1.2	main.c	2

## 1 ファイル詳解

### 1.1 main.c ファイル

文字列中の文字の有無を判定する

```
#include <stdio.h>
#include <stdbool.h>
```

関数

- bool [find\\_letter](#) (char x, char str[])  
文字列 *str* 中の文字 *x* の位置を返す
- int [main](#) (int argc, char \*argv[])

#### 1.1.1 詳解

文字列中の文字の有無を判定する

日付

2016/12/15

著者

佐伯雄飛  
B162392

[main.c](#) に定義があります。

#### 1.1.2 関数詳解

##### 1.1.2.1 bool find\_letter ( char x, char str[] )

文字列 *str* 中の文字 *x* の位置を返す

仕様：

- 文字列中に *x* が存在すれば true , なければ false

## 引数

<code>x</code>	検索する文字 (char 型)
<code>str</code>	検索対象の文字列 (char[] 型)

## 戻り値

bool 型 . `str` 中に `x` が存在すれば true , そうでなければ false

## 日付

2016/12/15

## 著者

佐伯雄飛 , B162392

`main.c` の行目に定義があります。

```

00024                                     {
00025     int i = 0;
00026     while (str[i] != '\0') {
00027         if (x == str[i]) {
00028             return true;
00029         }
00030         i++;
00031     }
00032     return false;
00033 }
```

## 1.1.2.2 int main ( int argc, char \* argv[] )

`main.c` の行目に定義があります。

```

00086                                     {
00087     // 引数の数を判定してください
00088     char x = *argv[1]; // コマンドライン引数から x に値を設定してください
00089     int len = 100;
00090
00091     char str[len];
00092     scanf("%99s", str);
00093     bool location = find_letter(x, str);
00094
00095     printf("%d\n", location ? 1 : 0);
00096
00097     return 0;
00098 }
```

## 1.2 main.c

```

00001 /** @file main.c
00002  * @brief 文字列中の文字の有無を判定する
00003  * @date 2016/12/15
00004  * @author 佐伯雄飛
00005  * @author B162392
00006 */
00007
00008 #include <stdio.h>
00009 #include <stdbool.h>
00010
00011 /** @fn bool find_letter(char x, char str[])
00012  * @brief 文字列 str 中の文字 x の位置を返す
00013  *
00014  * 仕様:
00015  * - 文字列中に x が存在すれば true , なければ false
00016  *
00017  * @param x 検索する文字 (char 型)
00018  * @param str 検索対象の文字列 (char[] 型)
00019  * @return bool 型 . str 中に x が存在すれば true , そうでなければ false
00020  * @date 2016/12/15
```

```
00021 * @author 佐伯雄飛, B162392
00022 */
00023
00024 bool find_letter(char x, char str[]) {
00025     int i = 0;
00026     while (str[i] != '\0') {
00027         if (x == str[i]) {
00028             return true;
00029         }
00030         i++;
00031     }
00032     return false;
00033 }
00034
00035 /** @fn int main(void)
00036 * @brief 文字列中の文字の有無を判定する
00037 *
00038 * 入力:
00039 * - コマンドライン引数に検索すべき文字 x が 1 つ与えられる
00040 * -
00041 * 標準入力には検索対象の、空白を含まない文字列 str が 1 つ与えられる。長さは len 以下。
00042 *
00043 * 出力:
00044 * - 文字列 str に文字 x が存在すれば 1 を、そうでなければ 0 を表示する。
00045 * - コマンドライン引数に x が与えられなければ、何も表示せずに終了する (return
00046 0 で)。
00047 *
00048 * 実行例:
00049 \verbatim
00050 ./main G
00051 \endverbatim
00052 * 入力例:
00053 \verbatim
00054 Hqb5GF&M3ilq8HlAyz7aL9OYjIbZXI1A
00055 \endverbatim
00056 * 出力例:
00057 \verbatim
00058 1
00059 \endverbatim
00060 * 実行例:
00061 \verbatim
00062 ./main X
00063 \endverbatim
00064 * 入力例:
00065 \verbatim
00066 JwaH
00067 \endverbatim
00068 * 出力例:
00069 \verbatim
00070 0
00071 \endverbatim
00072 * 実行例:
00073 \verbatim
00074 ./main
00075 \endverbatim
00076 * 入力例:
00077 \verbatim
00078 asflk;j;lfas
00079 \endverbatim
00080 * 出力例:
00081 \verbatim
00082 \endverbatim
00083 * @date 2016/12/15
00084 * @author 佐伯雄飛, B162392
00085 */
00086 int main(int argc, char* argv[]) {
00087     // 引数の数を判定してください
00088     char x = *argv[1]; // コマンドライン引数から x に値を設定してください
00089     int len = 100;
00090
00091     char str[len];
00092     scanf("%99s", str);
00093     bool location = find_letter(x, str);
00094
00095     printf("%d\n", location ? 1 : 0);
00096
00097     return 0;
00098 }
```