

# c2html user manual

---

First edition, for c2html version 1.0.0 Bèta  
Last updated March 21, 2004

```
#include <stdio.h>
#define MAX_ARGS 100
#define ERR_UNVALD_ARG 1
#define ERR_UNREC_ARG 2

int main (int argc, char *argv[])
{
    if (argc != MAX_ARGS) /* If the number of arguments is not correct */
    {
        printf ("ERROR: invalid number of arguments");
        exit (ERR_UNVALD_ARG);
    }
    if (*argv[1] != '-')
    {
        printf ("ERROR: unrecognized option '%s'", argv[1]);
        exit (ERR_UNREC_ARG);
    }
    switch (argv[1][1])
    {
        case 'b':
            printf ("Hi\n");
            break;
        case 'h':
            printf ("Hello world!\n");
            break;
        default:
            printf ("ERROR: unrecognized option '%s'", argv[1]);
            exit (ERR_UNREC_ARG);
            break;
    }
    exit (0);
}
```

c2html

# Table of Contents

1.What is c2html?.....	2
2.Running c2html.....	2
3.The output file.....	2
4.Possibilities.....	3
Highlighting.....	3
Tabs.....	3
Entities.....	3
HTML title.....	4
5.Example.....	4
Running c2html.....	4

## 1.What is c2html?

C2html is a program which converts C source files to highlighted html-files. The program has a commandline interface.

## 2.Running c2html

You can run c2html by typing the following at the commandline:

```
c2html [file]
```

or:

```
c2html [argument]
```

File is the name of the C source file to convert. For argument see table 1.

<i>Argument</i>	<i>Function</i>
-a	Print gernal information and exit
--help or -h	Print help and exit
--version	Print version information and exit

Table 1

## 3.The output file

The output file of the program is a html-file which will be written to the same directory as the input file (the C source file) is in. The name of it is the name of the input file plus the html file-extention (".html").

Example:

Show at the commandline the contents of the current directory:

```
ls
```

The following was printed to the screen:

```
main.c
```

Now we execute c2html with main.c as input file:

```
c2html main.c
```

C2html created a html file named "main.c.html" in the working directory.

Show again the contents of the current directory:

```
ls
```

The following was printed to the screen:

```
main.c      main.c.html
```

You see!

## 4.Possibilities

### Highlighting

As said before c2html converts a C source file to a highlighted html-file. The program distinguishes 11 different kinds of codeparts for highlighting. Every kind of codepart has a specific color and fontstyle. Table 2 shows the color and fontstyle for every kind of codepart.

<i>Codepart</i>	<i>Color</i>	<i>Fontstyle</i>	<i>Example</i>
Comment	grey	italic	<i>/* Comment */</i>
Keyword	black	bold	<b>for</b>
Data type	red	regular	char
Char	light magenta	regular	'A'
Decimal	light blue	regular	99
Float	magenta	regular	9.99
Hex	turquoise	regular	0xFF61
Octal	turquoise	regular	0156
Preprocessor	green	regular	#include <stdio.h>
String	light red	regular	"String"
Normal text	black	regular	count++;

Table 2

### Tabs

HTML does not support tabs (the '\t' character). Instead of writing a tab to the html-file, c2html writes a calculated number of (so called) non-beaking spaces. The program writes these spaces until it reaches a tab stop. About every 8 characters a tab stop occurs. A demonstration code to make this clear is shown below.

```
/* Every '|' is a tab stop
   the space before a tab stop is a tab ('\t')

0123456781234567812345678
      |   |   |   |
      aa s | jkw | s
*/
```

### Entities

Some characters (like '<') can not be used in a html code for representing text. For these characters c2html will replace them by a appropriate html entity. The '<'-character for

example will in the html-file be replaced by “&lt;”.

## ***HTML title***

The title of the html-file is equal to the name of the input file.

## **5.Example**

In this chapter we demonstrate with an example how to use c2html. In this example we have created a source file called main.c in /home/me/src/. These are the contents of main.c:

```
/*
   This is a demonstration C source file, used for the c2html user manual.
*/
#include <stdio.h>

#define NUM_ARGS          2
#define ERR_INVALID_NUM_ARGS  1
#define ERR_UNREC_ARG       2

int main (int argc, char *argv[])
{
    if (argc != NUM_ARGS) /* If the number of arguments is not correct */
    {
        printf ("ERROR: invallid number of arguments\n");
        exit (ERR_INVALID_NUM_ARGS);
    }
    if (*argv[1] != '-')
    {
        printf ("ERROR: unrecognized option '%s'\n", argv[1]);
        exit (ERR_UNREC_ARG);
    }
    switch (argv[1][1])
    {
        case 'b':
            printf ("Bla\n");
            break;
        case 'h':
            printf ("Hello world\n");
            break;
        default:
            printf ("ERROR: unrecognized option '%s'\n", argv[1]);
            exit (ERR_UNREC_ARG);
            break;
    }
    exit (0);
}
```

## ***Running c2html***

We execute c2html with main.c as input file:

```
c2html main.c
```

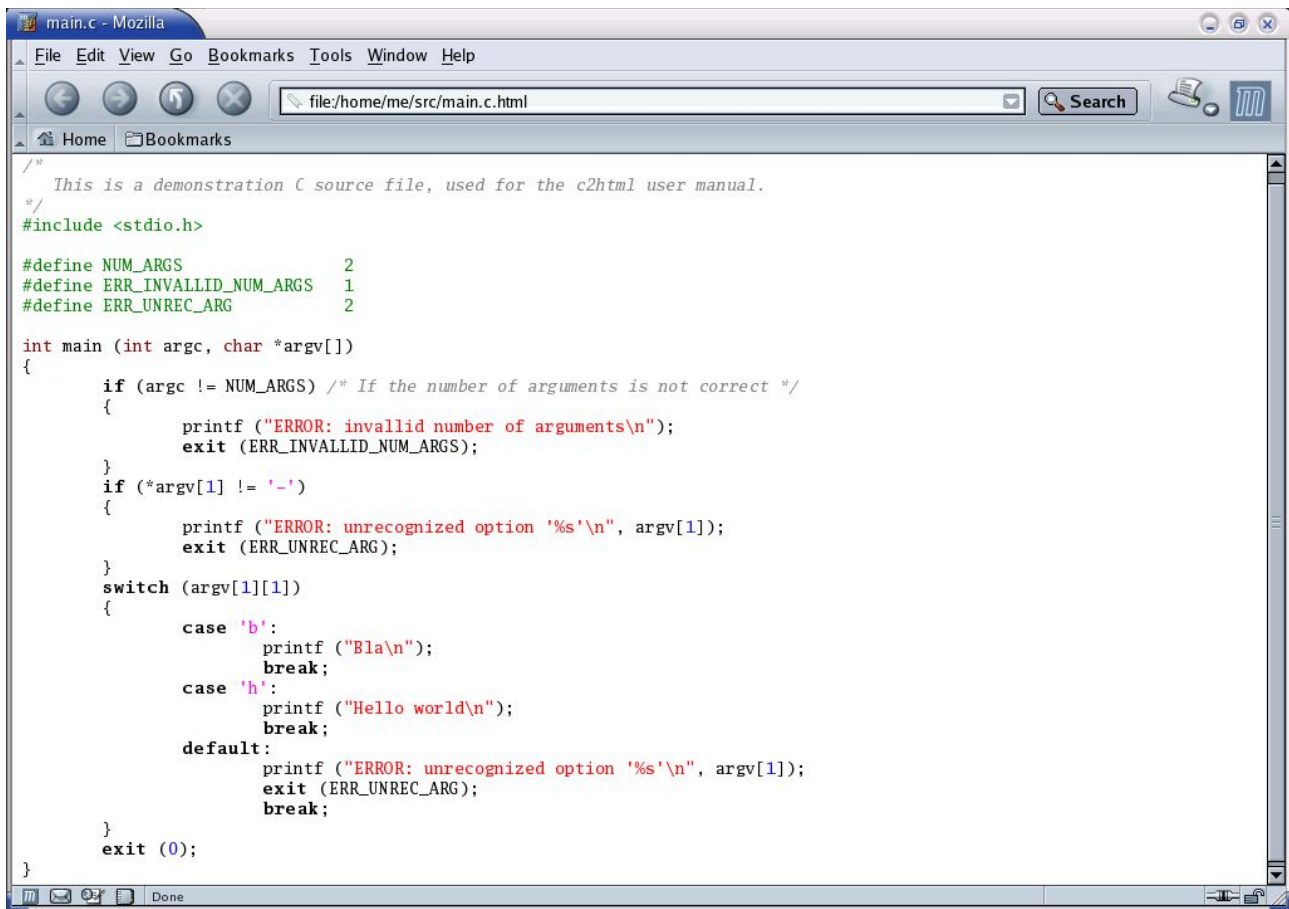
Show the contents of /home/me/src (current directory):

```
ls
```

You can see c2html made a html-file:

main.c	main.c.html
--------	-------------

When we open main.c.html in a webbrowser (see illustration 1), we see a beautiful highlighted C source code.



```
main.c - Mozilla
File Edit View Go Bookmarks Tools Window Help
file:/home/me/src/main.c.html Search
Home Bookmarks
/*
 * This is a demonstration C source file, used for the c2html user manual.
 */
#include <stdio.h>

#define NUM_ARGS      2
#define ERR_INVALID_NUM_ARGS  1
#define ERR_UNREC_ARG  2

int main (int argc, char *argv[])
{
    if (argc != NUM_ARGS) /* If the number of arguments is not correct */
    {
        printf ("ERROR: invalid number of arguments\n");
        exit (ERR_INVALID_NUM_ARGS);
    }
    if (*argv[1] != '-')
    {
        printf ("ERROR: unrecognized option '%s'\n", argv[1]);
        exit (ERR_UNREC_ARG);
    }
    switch (argv[1][1])
    {
        case 'b':
            printf ("Bla\n");
            break;
        case 'h':
            printf ("Hello world\n");
            break;
        default:
            printf ("ERROR: unrecognized option '%s'\n", argv[1]);
            exit (ERR_UNREC_ARG);
            break;
    }
    exit (0);
}
```

Illustration 1: *main.c.html, created by c2html, opened in Mozilla*