4.5 – Trigraph characters in C language

by subbu on September 12, 2013

Why trigraph characters wer e used?

In C language we use different characters on different purposes. Say for example {} are used to define a scope, [] are used to define dimension to an array, \ is used to write any escape sequence, # is an initial character to any pre-processor statement, | is a bitwise OR, ^ is a bitwise XOR and ~ is a negation operator.

Keyboards used with some platforms were missing some of these characters. So it was not possible to write C code on these machines.

Trigraph characters:

- Some of the characters like {}, [], \, |, ~ and ^ are missing in the above keyboard. Hence practically it may not be possible to write a C program using this keyboard.
- To solve this problem C suggested to use combination of 3 characters to produce a single character called trigraph character.
- A trigraph is a sequence of three characters, the first two of which are question marks
- C supports the following 9 trigraph characters.

Trigraph sequence	Equal character
??=	#
??(	[
??)	]
??/	\
??<	{
??>	}
??!	|
??’	^
??-	~

- The *trigraphs preprocessor* replaces all occurrences of trigraph sequences by their single-character equivalents before any other processing.

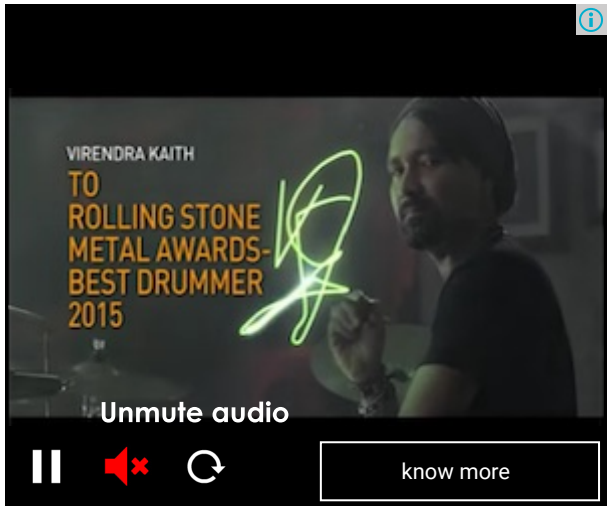
We need to write a simple hello world program in the following way in the platform which doesn't support #, {} and \ characters.

```
??=include<stdio.h>
int main()
??<
printf("Hello??/nWorld");
return 0;
??>
```

The trigraphs pre-processor changes the above code as

```
#include<stdio.h>
int main()
{
printf("Hello\nWorld");
return 0;
}
```

Trigraphs are not commonly supported by all the compilers. Some compilers support an option to turn recognition of trigraphs on. Some issues warnings when they encounter trigraphs in the source files



Executing a C program with trigraph characters in Turbo C:

Turbo C provides a trigraph preprocessor called **trigraph**, which accepts a C source file as input and replaces all the trigraph characters with their associated characters. In simple, trigraph preprocessor changes the source file.

The trigraph preprocessor is available in the Turbo C bin folder. Make sure that, the path is set to the bin folder of Turbo C as specified in 1.6 How to execute a C program in command prompt using Turbo C session to access the trigraph tool from any where

Once trigraph preprocessor replaces all the trigraph characters with their appropriate characters in the source then C program can be compiled and executed as a normal program using tcc

1. Open the text editor to write the program

```
>edit prog.c
```

2. Type the program, save (alt+f, s) and exit (alt+f, x) from the editor

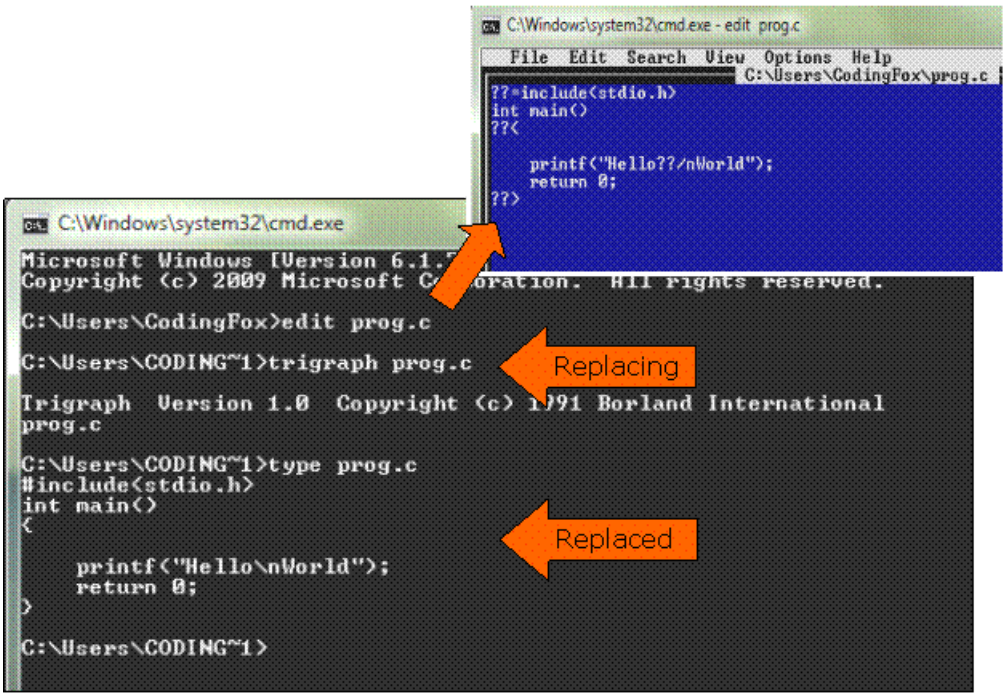
```
/* program to print hello world */
??=include<stdio.h>
int main()
??<
printf("Hello??/nWorld");
return 0;
??>
```

3. Replace all the trigraph characters in the source code (prog.c) using trigraph preprocessor

```
>trigraph prog.c
```

4. We can see the replaced source file using the **type** command

```
>type prog.c
#include<stdio.h>
int main()
{
printf("Hello\nWorld");
return 0;
}
```

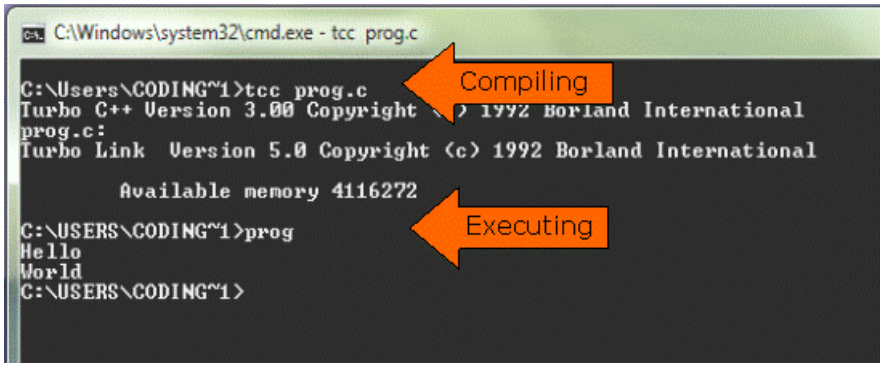


5. Compile the program using tcc (Turbo C compiler)

```
>tcc prog.c
```

6. Execute the program

```
>prog
Hello
World
```



Executing a C program with trigraph characters in windows:

Now we will see how a C program with trigraphs is executed using MinGw gcc compiler on windows command prompt.

1. Open the text editor to write the program

```
>edit hello.c
```

2. Type the program, save (alt+f, s) and exit (alt+f, x) from the editor

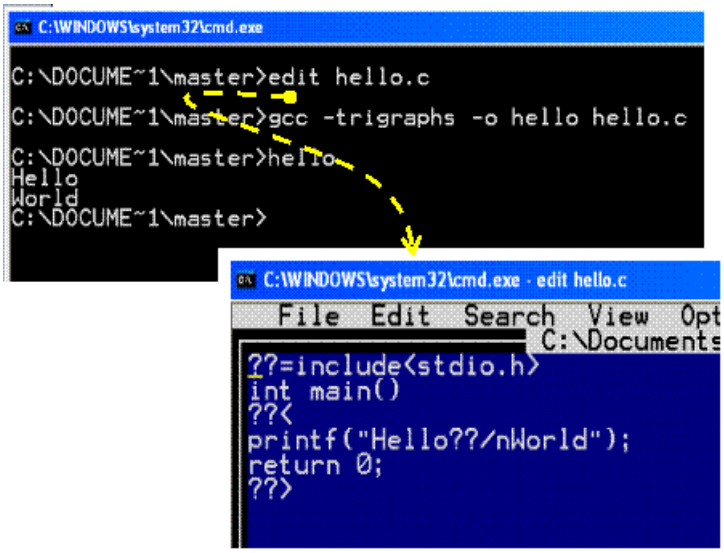
```
1  /* program to print hello world */
2  ??=include<stdio.h>
3  int main()
4  ??<
5  printf("Hello??/nWorld");
6  return 0;
7  ??>
```

3. Compile the program using “gcc”.
Note:
-trigraphs is a compiler option to enable trigraph preprocessor
-o is a compiler option to generate executable file with file name (hello.exe)

```
>gcc -trigraphs -o hello hello.c
```

4. Execute the program

```
>hello
Hello
World
```



Executing a C program with trigraph characters in Linux:

Now we will see how a C program with trigraphs is executed using gcc compiler in Linux.

1. Open the gedit to write the C program

```
$gedit sum.c
```

2. Type the program, save (ctrl+s) and exit (ctrl+q) from the editor

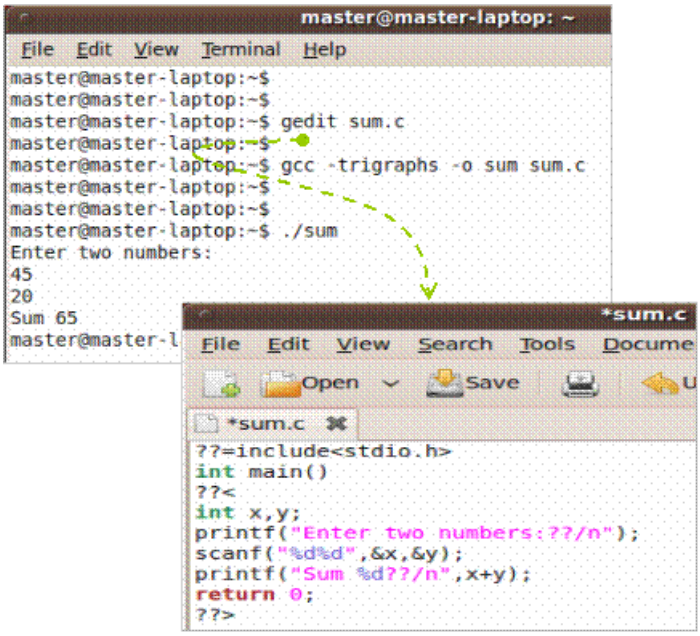
```
1  /* program to take two numbers and print sum */
2  ??=include<stdio.h>
3  int main()
4  ??<
5  int x,y;
6  printf("Enter two numbers:??/n");
7  scanf("%d%d",&x,&y);
8  printf("Sum %d??/n",x+y);
9  return 0;
10 ??>
```

3. Compile the program using gcc

```
$gcc -trigraphs -o sum sum.c
```

4. Execute the program

```
$./sum
Enter two numbers:
45
20
Sum 65
```



Conclusion:

Almost all the keyboards and platforms now a days support the standard character set, hence we no need to use trigraph characters in any C program

Previous post: 4.3 – Escape sequences or Back slash characters

Next post: 4.6 – Digraph characters in C language

- **C Home Page**

- **LATEST**

  - 1.3.2 – Executing java program in NetBeans
  - 1.3.1 – Installation of Net Beans on Windows
  - 1.2.2 – Executing a Java Program in Edit+
  - 1.2.1 – Configuring Edit+ for java
  - 1.1.3 – Executing a java program from command prompt (Windows)
  - 1.1.2 – Configuring JDK on Windows
  - 1.1.1 – Installation of JDK on Windows
  - 0.8 – JVM vs JRE vs JDK
  - 0.7 – Java Virtual Machine and Java Runtime Environment
  - 0.6 – Java as a platform independent language

- **SEARCH TOPIC**

- **Archive**

  - Dec 2014 (7)
  - Nov 2014 (4)
  - Oct 2014 (20)
  - Sep 2014 (14)
  - Aug 2014 (7)
  - Feb 2014 (5)
  - Jan 2014 (27)
  - Dec 2013 (29)
  - Nov 2013 (23)
  - Oct 2013 (22)
  - Sep 2013 (36)
  - Aug 2013 (17)
  - Jul 2013 (16)

Rs.947
NMD-330
GSM-ID-Card
Rs.7,999
Rs.10,000
Rs.169
Delhi 6
ONLINE
Rs.199
Rs.9,999
Rs.99
LED
BULB
Rs.99
Rs.375
GSM
Rs.1,800
Rs.399
Rs.9,999
Rs.9,999
Rs.269
SHOPCLUES.COM

Rs.9,999

Rs.947

Rs.8,999

SHOPCLUES.COM

This site is designed for teaching computer programming in simple, easy, complete and clearer way. Examples are simplified to improve understanding and reviewed constantly to avoid errors, but can't warrant on full correctness.
Copyright©2013 by CodingFox.com. All rights reserved.

codingfox.com