



Аргументы командной строки в C с использованием argc и argv



Автор: Алекс Аллейн

В C можно принимать аргументы командной строки. Аргументы командной строки задаются после имени программы в операционных системах командной строки, таких как DOS или Linux, и передаются программе из операционной системы. Чтобы использовать аргументы командной строки в своей программе, вы должны сначала понять полное объявление функции main, которая ранее не принимала аргументов. На самом деле main может принимать два аргумента: один аргумент-количество аргументов командной строки, а другой - полный список всех аргументов командной строки.

Полная декларация main выглядит следующим образом:

```
1 | int main ( int argc, char *argv[] )
```

Целое число, argc-это **аргументс**count. Это количество аргументов, передаваемых в программу из командной строки, включая имя программы.

Массив символьных указателей - это список всех аргументов. argv[0] - это имя программы или пустая строка, если это имя недоступно. После этого каждый номер элемента, меньший argc, является аргументом командной строки. Вы можете использовать каждый элемент argv как строку или использовать argv как двумерный массив. argv[argc] - это нулевой указатель.

Как это можно использовать? Почти любая программа, которая хочет, чтобы ее параметры были установлены при выполнении, будет использовать это. Одним из распространенных способов использования является написание функции, которая принимает имя файла и выводит весь его текст на экран.

```
1 | #include <stdio.h>
2 |
3 | int main ( int argc, char *argv[] )
4 | {
5 |     if ( argc != 2 ) /* argc should be
6 |     {
7 |         /* We print argv[0] assuming it
8 |         printf( "usage: %s filename",
9 |     }
10 |    else
11 |    {
12 |        // We assume argv[1] is a file
13 |        FILE *file = fopen( argv[1], '
14 |
15 |        /* fopen returns 0, the NULL p
16 |        if ( file == 0 )
17 |        {
18 |            printf( "Could not open fi
19 |        }
20 |        else
21 |        {
22 |            int x;
23 |            /* read one character at a
24 |            indicates the end of th
25 |            to a variable, check th
26 |            the assignment statemen
27 |            while ( ( x = fgetc( file
28 |            {
29 |                printf( "%c", x );
30 |            }
31 |            fclose( file );
32 |        }
33 |    }
34 | }
```

Эта программа довольно короткая, но она включает в себя полную версию main и даже выполняет полезную функцию. Сначала он проверяет, добавил ли пользователь второй аргумент, теоретически имя файла. Затем программа проверяет, действителен ли файл, пытаясь открыть его. Это стандартная операция, и если она приводит к открытию файла, то возвращаемое значение fopen будет допустимым **ФАЙЛОМ***; в противном случае оно будет равно 0, нулевому указателю. После этого мы просто выполняем цикл, чтобы распечатать из файла по одному символу за раз. Код таков: он говорит сам за себя, но завален комментариями; у вас не должно возникнуть проблем с пониманием его работы так далеко в учебнике. :-)

Проверьте себя

Предыдущий: Типизация

Следующий: Связанные списки

Обратно в индекс учебника C

Реклама | Политика конфиденциальности | Copyright © 2019 Cprogramming.com | Контакты | О компании