We do not currently allow content pasted from ChatGPT on Stack Overflow; read our policy here.

Input from the execution line in the terminal in c

Asked 11 years ago Modified 2 years, 4 months ago Viewed 14k times



The problem that i have is that i have to write a hanois tower game in c and the input for the number of the rings must not be in the programm but the code must read the number of rings in the execution.



0

Example: ./hanoistower 3



And the code should get the 3 as the input. How can i do that?



terminal line

Share Improve this question Follow

edited Jul 10, 2013 at 17:16 user1228

asked Nov 30, 2011 at 9:58



3 Answers



Highest score (default)



Command line arguments are propagated as strings through the main() function of your C program.





In int main(int argc, char *argv[]) argc is the number of arguments, and argv is an array of strings containing the arguments. Note that the program name itself is always the first "argument".



As the arguments are passed as strings, you likely need to convert your 3 to an integer, which can be done with the atoi function. Here's a start:



```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[])
   int rings;
   if(argc != 2) {
       printf("Usage: %s number-of-rings\n",argv[0]);
```

Join Stack Overflow to find the best answer to your technical question, help others answer theirs.

Sign up



```
printf("Using number-of-rings = %d\n", rings);
...
return 0;
}
```

Share Improve this answer Follow

answered Nov 30, 2011 at 10:03



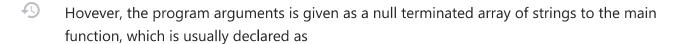
nos

219k 56 407 496



I strongly suggest reading a good C programming book (in 2020, Modern C).

It will be much faster than asking questions here. Don't forget to also read the documentation of your C compiler (perhaps GCC), your <u>build automation</u> tool (e.g. <u>GNU make</u> or <u>ninja</u>), and debugger (perhaps <u>GDB</u>). If you code on and for Linux, read also <u>Advanced Linux Programming</u> and <u>syscalls(2)</u>, and the documentation of <u>GNU glibc</u> (or of <u>musl-libc</u>, if you use it).



```
int main (int argc, char**argv) { /*...*/ }
```

if you run your program with ./hanoistower 3 and if your hanoistower.c is your source code (which you need to compile with debugging and warning enabled, i.e. gcc -Wall -g hanoistower.c -o hanoistower on Linux) then you have one extra argument, so

- 1. argc == 2
- argv[0] is the "./hanoistower" string
- 3. argv[1] is the "2" string (use atoi to convert it to an int)
- 4. argv[2] is NULL

Share Improve this answer Follow

Please, please learn to use the debugger (gdb on Linux).

answered Nov 30, 2011 at 10:07



Basile Starynkevitch

I'm very surprised that you find simpler to ask on a forum than to go to a library and take time (several hours) to read a good C programming book. – Basile Starynkevitch Nov 30, 2011 at 10:30

You have no library at your university? You know, paper textbooks!! – Basile Starynkevitch Nov 30, 2011 at 11:02

edited Aug 7, 2020 at 17:44

Join Stack Overflow to find the best answer to your technical question, help others answer theirs.

Sign up





Just add, argc and argv to the list of main method parameters, as shown below:



int main (int argc, char *argv[])



Then use argy as the variable to specify number of rings inside your code.



Share Improve this answer Follow

edited Nov 30, 2011 at 10:16



answered Nov 30, 2011 at 10:08



Join Stack Overflow to find the best answer to your technical question, help others answer theirs.

Sign up