

Tip 7: Include header files from the command line

Ben Klemens

15 October 2011

[A tip every other day on POSIX and C. Start from the tip intro page¹.]

level: library user

purpose: really never think about standard headers again

The GCC has a convenient flag for including headers:

```
gcc -include stdio.h
```

is equivalent to putting

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

at the head of your C file.

By adding that to our invocation of GCC, we can finally write `hello.c` as the one line of code it should be:

```
int main(){ printf("Hello, world.\n"); }
```

which compiles fine via:

```
gcc -include stdio.h hello.c -o hi
```

To do:

In tip #5, you wrote a single header file to join them all, so that you could just write

```
#include <allheads.h>
```

at the top of your program. Now you don't even need to do that: modify the makefile from tip #1 by adding `-include allheads.h` on the `CFLAGS` line.

Discussion:

My impression is that C programmers hold themselves to a much higher standard of portability than people who write code in the typical scripting language. With no standardized environment by a central authority, it takes more discipline to ensure that everything will compile everywhere.

¹<http://modelingwithdata.org/arch/00000049.htm>

This tip bucks that tendency a bit, because we're saying the program will only compile correctly if you give just the right flag to the compiler (and this is even GCC specific). If this bothers you, well, skip this tip.

I'm not bothered for two reasons. First, I always attach a makefile with every program I send to anybody, because that significantly raises the odds that the other side will be able to get the program running. Second, most of what I write will never see the light of day, but is written for my own research or idiosyncratic needs. For the one-in-a-hundred stupid little C scripts for my personal use that grow into something used by others, I might take the time to explicitly `#include <apop.h>` rather than using the `-include apop.h` flag from this tip.