Where are the constructor and destructor attribute directives for Clang documented?

Function attributes for C and C++ for running code before and after `main()` work with the Clang compiler. The following code:

```
#include <stdio.h>

void constr() __attribute((constructor));
void destr() __attribute((destructor));

int main(){
    printf("foo\n");
    return 0;
}

void constr(){
    printf("bar\n");
```

```
void destr(){
    printf("baz\n");
}
```

Works as expected:

```
clang++ const_dest.cpp -o const_dest; ./const_dest
```

Outputs:

```
bar
foo
baz
```

However, unlike GCC, I can't find such a description in the official documentation. This page lists numerous attributes, but constructor and destructor are not among them. A google search of

```
"__attribute__ ((constructor))" site:llvm.org
```

(note the quotes) yields no result.

So, is this feature of Clang documented, and if not, why not? Seems a pretty strange omission, especially since a page listing many other attributes exists.

c++ c attributes documentation clang

asked Mar 21 at 21:44

executor21

3,173 5 21 32

1 Answer

The Clang documentation page you linked to says

> This page lists the attributes currently supported by Clang.

currently supported by the documented version (5) of Clang than as asserting that it documents only some of them.

You ask

> So, is this feature of Clang documented, and if not, why not? Seems a pretty strange omission, especially since a page listing many other attributes exists.

Either the attributes you asked about are *not* supported by Clang 5, or the documentation is faulty. I would believe either, or even both. Note, however, that "supported" is not necessarily the same thing as "recognized" or "accepted" or even "implemented". It may be that Clang accepts them but does nothing with them, or that it in fact implements them consistently with GCC, but does not support their use. One could even interpret "supported" in that context to mean "documented on this page."

answered Mar 21 at 21:54

John Bollinger

60.9k 5 26 62