

# Re: some C questions

---

*Source:* [http://coding.derkeiler.com/Archive/C\\_CPP/comp.lang.c/2007-05/msg02945.html](http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2007-05/msg02945.html)

---

- *From:* Chris Hills <chris@xxxxxxxxxxxx>
  - *Date:* Tue, 29 May 2007 13:04:19 +0100
- 

In article <1180432199.553430.252930@xx>, Guru Jois <guru.jois@xxxxxxxx> writes

On May 24, 2:21 pm, John Bode <john\_b...@xxxxxxxxxxxx> wrote:

On Wed, 23 May 2007 02:17:28 -0700, Guru Jois wrote:

> Hai all,

> I havesomequestion. Please answer.

> 1. What is walking pointer?

If you're asking what I think you're asking, it's the process of iterating through an array by incrementing a pointer, as below:

```
void copy(char *src, char *dst)
{
while(*src)
*dst++ = *src++;
*dst = 0;
}
```

Basically, we're "walking" through the src and dst arrays by incrementing the pointers.

> 2. What is difference between procedure and subroutine?

Practically speaking, none. Somelanguages such as Fortran and Ada distinguish between subroutines that return a value (functions) vs. subroutines that don't return a value (procedures) with different syntax and semantics (e.g., using different keywords to define the subroutine, or preventing you from modifying input parameters of a function). Cdoesn't make this distinction; all subroutines are functions.

> 3.What is template of main inC?

The standard prototypes for main are:

## Re: some C questions

```
int main(void);  
int main(int argc, char **argv);
```

Individual implementations may define additional prototypes, but must support at least those two.

> 4. What is padding of structure ( or structure padding )?

Most architectures require that multibyte objects (ints, floats, doubles, etc.) start on addresses that are multiples of 2 or 4. If you have a struct that has a single char member followed by an int member, the compiler will "pad" the struct so that the int member starts on the next even address, so you wind up with a dummy byte between the members.

> 5. What is advantages of using pointers to functions?.

There are a number of advantages, from allowing for plug-in architectures to introducing basic polymorphism. Unfortunately, I have to run out of here before I can come up with any decent examples. It's something that won't make sense until you've done a little more programming, anyway.

GOOD SUGGESTION COMPARED TO ALL OTHERS.

Thank you John Bode

I hope you should assist everyone to how to answer for the questions.

No. Many of us know how to answer these questions. But we don't do home work.

Giving irrelevant answers doesn't mirrors their expertness in C. If you all find my question irrelevant, why didn't John Bode??  
Hope you follow good path of Mr Bode.

Bye  
Guru Jois

From Guru's email to me re a book on compiler design....

- > Mr Hills,
- > Thanks for your suggestion. I now realize I don't need that book. Any
- > way can I know in which company you are working?. I am Master of
- > Computer Application graduate and currently studying "Advanced C and
- > Unix" course in Uttara, Bangalore, India.
- > Have you heard about UTTARA in Bangalore?
- >
- > Bye
- > Guru Jois

