C Pointers Interview Questions And Answers Guide.

Global Guideline.

https://globalguideline.com/

C Pointers Job Interview Preparation Guide.

Question # 1

Explain indirection?

Answer:-

If you declare a variable, its name is a direct reference to its value. If you have a pointer to a variable, or any other object in memory, you have an indirect reference to its value. If p is a pointer, the value of p is the address of the object. *p means "apply the indirection operator to p"; its value is the value of the object that p points to. (Some people would read it as "Go indirect on p.")

Read More Answers.

Question # 2

How many levels of pointers have?

Answer:-

The answer depends on what you mean by "levels of pointers." If you mean "How many levels of indirection can you have in a single declaration?" the answer is "At least 12."

Read More Answers.

Question # 3

For what purpose null pointer used?

Answer:-

The null pointer is used in three ways:
* To stop indirection in a recursive data structure.
* As an error value.
* As a sentinel value.

Read More Answers.

Question # 4

Tell me when is a void pointer used?

Answer:-

A void pointer is used for working with raw memory or for passing a pointer to an unspecified type.
Some C code operates on raw memory. When C was first invented, character pointers (char *) were used for that. Then people started getting confused about when a character pointer was a string, when it was a character array, and when it was raw memory.

Read More Answers.

Question # 5

Tell me is NULL always defined as 0(zero)?

Answer:-

NULL is defined as either 0 or (void*)0. These values are almost identical; either a literal zero or a void pointer is converted automatically to any kind of pointer, as necessary, whenever a pointer is needed (although the compiler can't always tell when a pointer is needed).

Read More Answers.

Question # 6

Can we add pointers together?

Answer:-

No, you can't add pointers together. If you live at 1332 Lakeview Drive, and your neighbor lives at 1364 Lakeview, what's 1332+1364? It's a number, but it doesn't mean anything. If you try to perform this type of calculation with pointers in a C program, your compiler will complain.

Read More Answers.

Question # 7

Tell me when would you use a pointer to a function?

Answer:-

Pointers to functions are interesting when you pass them to other functions. A function that takes function pointers says, in effect, "Part of what I do can be customized. Give me a pointer to a function, and I'll call it when that part of the job needs to be done. That function can do its part for me." This is known as a "callback." It's used a lot in graphical user interface libraries, in which the style of a display is built into the library but the contents of the display are part of the application.

Read More Answers.

Question # 8

Tell me can the size of an array be declared at runtime?

Answer:-

No. In an array declaration, the size must be known at compile time. You can't specify a size that's known only at runtime.

Read More Answers.

Question # 9

What is (void*)0?
A. Representation of NULL pointer
B. Representation of void pointer
C. Error
D. None of above

Answer:-

Option A
(Representation of NULL pointer)

Read More Answers.

Question # 10

Can you combine the following two statements into one?
char *p;
p = (char*) malloc(100);
A. char p = *malloc(100);
B. char *p = (char) malloc(100);
C. char *p = (char*)malloc(100);
D. char *p = (char *)(malloc*)(100);

Answer:-

Option C
(char *p = (char*)malloc(100);)

Read More Answers.

Question # 11

In which header file is the NULL macro defined?
A. stdio.h
B. stddef.h
C. stdio.h and stddef.h
D. math.h

Answer:-

Option C
(stdio.h and stddef.h)

Read More Answers.

Question # 12

How many bytes are occupied by near, far and huge pointers (DOS)?
A. near=2 far=4 huge=4
B. near=4 far=8 huge=8
C. near=2 far=4 huge=8
D. near=4 far=4 huge=8

Answer:-

Option A
(near=2 far=4 huge=4)

Read More Answers.

Question # 13

If a variable is a pointer to a structure, then which of the following operator is used to access data members of the structure through the pointer variable?
A. .
B. &
C. *
D. ->

Answer:-

Option D
(->)

Read More Answers.

Question # 14

What would be the equivalent pointer expression for referring the array element a[i][j][k][l]
A. ((((a+i)+j)+k)+l)
B. *(*(*(*(a+i)+j)+k)+l)
C. (((a+i)+j)+k+l)
D. ((a+i)+j+k+l)

Answer:-

Option B
((*(*(a+i)+j)+k)+l)

Read More Answers.

Question # 15

A pointer is
A. A keyword used to create variables
B. A variable that stores address of an instruction
C. A variable that stores address of other variable
D. All of the above

Answer:-

Option C
(A variable that stores address of other variable)

Read More Answers.

Question # 16

The operator used to get value at address stored in a pointer variable is
A. *
B. &
C. &&
D. ||

Answer:-

Option A
(*)

Read More Answers.

Question # 17

Explain void pointer?

Answer:-

A void pointer is a C convention for "a raw address." The compiler has no idea what type of object a void pointer "really points to." If you write
int *ip;
ip points to an int. If you write
void *p;
p doesn't point to a void!

Read More Answers.

Question # 18

Explain null pointer?

Answer:-

There are times when it's necessary to have a pointer that doesn't point to anything. The macro NULL, defined in <stddef.h>, has a value that's guaranteed to be different from any valid pointer. NULL is a literal zero, possibly cast to void* or char*. Some people, notably C++ programmers, prefer to use 0 rather than NULL.<stddef.h>

Read More Answers.

Question # 19

Do you know the use of fflush() function?

Answer:-

In ANSI, fflush() [returns 0 if buffer successfully deleted / returns EOF on an error] causes the system to empty the buffer associated with the specified output stream.
It undoes the effect of any ungetc() function if the stream is open for input. The stream remains open after the call.
If stream is NULL, the system flushes all open streams.
However, the system automatically deletes buffers when the stream is closed or even when a program ends normally without closing the stream.

Read More Answers.

Question # 20

Do you know NULL pointer?

Answer:-

A null pointer does not point to any object.
NULL and 0 are interchangeable in pointer contexts.Usage of NULL should be considered a gentle reminder that a pointer is involved.
It is only in pointer contexts that NULL and 0 are equivalent. NULL should not be used when another kind of 0 is required.

Read More Answers.

Question # 21

What is the difference between exit() and _exit() function?

Answer:-

The following are the differences between exit() and _exit() functions:
- io buffers are flushed by exit() and executes some functions those are registered by atexit().
- _exit() ends the process without invoking the functions which are registered by atexit().

Read More Answers.

Question # 22

What are the advantages of using macro in C Language?

Answer:-

In modular programming, using functions is advisable when a certain code is repeated several times in a program. However, everytime a function is called the control gets transferred to that function and then back to the calling function. This consumes a lot of execution time. One way to save this time is by using macros. Macros substitute a function call by the definition of that function. This saves execution time to a great extent.

Read More Answers.

Question # 23

Explain what is the purpose of "extern" keyword in a function declaration?

Answer:-

The declaration of functions defaults to external linkage. The only other storage class possible for a function is static, which must be specified explicitly. It cannot be applied to a block scope function declaration and results in internal linkage.

Read More Answers.

Question # 24

What is the difference between strcpy() and memcpy() function?

Answer:-

The following are the differences between strcpy() and memcpy():
- memcpy() copies specific number of bytes from source to destinatio in RAM, where as strcpy() copies a constant / string into another string.
- memcpy() works on fixed length of arbitrary data, where as strcpy() works on null-terminated strings and it has no length limitations.
- memcpy() is used to copy the exact amount of data, whereas strcpy() is used of copy variable-length null terminated strings.

Read More Answers.

Question # 25

Explain #pragma statements?

Answer:-

The #pragma Directives are used to turn ON or OFF certain features. They vary from compiler to compiler.
Examples of pragmas are:
#pragma startup // you can use this to execute a function at startup of a program
#pragma exit // you can use this to execute a function at exiting of a program
#pragma warn â€"rvl // used to suppress return value not used warning
#pragma warn â€"par // used to suppress parameter not used warning
#pragma warn â€"rch // used to suppress unreachable code warning

Read More Answers.

Question # 26

Do you know the difference between malloc() and calloc() function?

Answer:-

The following are the differences between malloc() and calloc():
- Byte of memory is allocated by malloc(), whereas block of memory is allocated by calloc().
- malloc() takes a single argument, the size of memory, where as calloc takes two parameters, the number of variables to allocate memory and size of bytes of a single variable
- Memory initialization is not performed by malloc() , whereas memory is initialized by calloc().
- malloc(s) returns a pointer with enough storage with s bytes, where as calloc(n,s) returns a pointer with enough contiguous storage each with s bytes.

Read More Answers.

Question # 27

Can you please compare array with pointer?

Answer:-

A pointer is an address location of another variable. It is a value that designates the address or memory location of some other value (usually value of a variable). The value of a variable can be accessed by a pointer which points to that variable. To do so, the reference operator (&) is pre-appended to the variable that holds the value. A pointer can hold any data type, including functions.

Read More Answers.

Question # 28

Tell me what are bitwise shift operators?

Answer:-

<< - Bitwise Left-Shift
Bitwise Left-Shift is useful when to want to MULTIPLY an integer (not floating point numbers) by a power of 2.
Expression: a << b
This expression returns the value of a multiplied by 2 to the power of b.
>> - Bitwise Right-Shift
Bitwise Right-Shift does the opposite, and takes away bits on the right.
Expression: a >> b
This expression returns the value of a divided by 2 to the power of b.

Read More Answers.

Question # 29

What is the use of bit field?

Answer:-

Packing of data in a structured format is allowed by using bit fields. When the memory is a premium, bit fields are extremely useful. For example:
- Picking multiple objects into a machine word : 1 bit flags can be compacted
- Reading external file formats : non-standard file formats could be read in, like 9 bit integers
This type of operations is supported in C language. This is achieved by putting :'bit length' after the variable. Example:

```
struct packed_struct {
      unsigned int var1:1;
      unsigned int var2:1;
      unsigned int var3:1;
      unsigned int var4:1;
      unsigned int var5:4;
      unsigned int funny_int:9;
} pack;
```

packed-struct has 6 members: four of 1 bit flags each, and 1 4 bit type and 1 9 bit funny_int.
C packs the bit fields in the structure automatically, as compactly as possible, which provides the maximum length of the field is less than or equal to the integer word length the computer system.
The following points need to be noted while working with bit fields:
The conversion of bit fields is always integer type for computation
Normal types and bit fields could be mixed / combined
Unsigned definitions are important.

Read More Answers.

Question # 30

What is the C Language function prototype?

Answer:-

The basic definition of a function is known as function prototype. The signature of the function is same that of a normal function, except instead of containing code, it always ends with semicolon.
When the compiler makes a single pass over each and every file that is compiled. If a function call is encountered by the compiler, which is not yet been defined, the compiler throws an error.
One of the solution for the above is to restructure the program, in which all the functions appear only before they are called in another function.
Another solution is writing the function prototypes at the beginning of the file, which ensures the C compiler to read and process the function definitions, before there is a change of calling the function. If prototypes are declared, it is convenient and comfortable for the developer to write the code of those functions which are just the needed ones.

Read More Answers.

Question # 31

Tell me with an example the self-referential structure?

Answer:-

A self referential structure is used to create data structures like linked lists, stacks, etc. Following is an example of this kind of structure:

```
struct struct_name
{
      datatype datatypename;
      struct_name * pointer_name;
};
```

Read More Answers.

C Language Most Popular Interview Topics.

1 : C Functions Frequently Asked Interview Questions and Answers Guide.

2 : C Preprocessor Frequently Asked Interview Questions and Answers Guide.

About Global Guideline.

Global Guideline is a platform to develop your own skills with thousands of job interview questions and web tutorials for fresher's and experienced candidates. These interview questions and web tutorials will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts. Global Guideline invite you to unlock your potentials with thousands of **Interview Questions with Answers** and much more. Learn the most common technologies at Global Guideline. We will help you to explore the resources of the World Wide Web and develop your own skills from the basics to the advanced. Here you will learn anything quite easily and you will really enjoy while learning. Global Guideline will help you to become a professional and Expert, well prepared for the future.

* This PDF was generated from https://GlobalGuideline.com at **November 29th, 2023**

* If any answer or question is incorrect or inappropriate or you have correct answer or you found any problem in this document then don't hesitate feel free and e-mail us we will fix it.

You can follow us on FaceBook for latest Jobs, Updates and other interviews material.
www.facebook.com/InterviewQuestionsAnswers

Follow us on Twitter for latest Jobs and interview preparation guides
https://twitter.com/InterviewGuide

Best Of Luck.

Global Guideline Team
https://GlobalGuideline.com
Info@globalguideline.com