

C | Pointer Basics | Question 17

```
#include <stdio.h>
void f(char**);
int main()
{
    char *argv[] = { "ab", "cd", "ef", "gh", "ij", "kl" };
    f(argv);
    return 0;
}
void f(char **p)
{
    char *t;
    t = (p += sizeof(int))[-1];
    printf("%s\n", t);
}
```

[Run on IDE](#)

- (A) ab
- (B) cd
- (C) ef
- (D) gh

Answer: (D)

Explanation:

[Quiz of this Question](#)



Choose from Top 150+
IT Courses & Rise in Your Career!

Big Data, Business Intelligence, Testing, Salesforce,
Data Science, DataBase, etc...

FLAT 15% OFF
COUPON CODE: ID15

 **Enroll Now**

See [Placement Course](#) for placement preparation, [GATE Corner](#) for GATE CS Preparation and [Quiz Corner](#) for all Quizzes on GeeksQuiz.

Category: C Tags: C, C-Pointer Basics, Pointer Basics

([Login](#) to Rate and Mark)

Average Rating : **0/5.0**

No votes yet.

★★★★★

Average Difficulty : **4/5.0**

4

Based on **1** vote(s)



Add to TODO List



Mark as DONE

11 Comments

GeeksQuiz

1 Login ▾

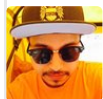
♥ Recommend

🔗 Share

Sort by Best ▾



Join the discussion...



Mr. Lazy • a year ago

Its simple ... check this ..

We know compiler converts `arr[i]` to `*(arr + i)` .

so `t = (p += sizeof(int))[-1]`; is converted to

`=> t = (p + 4)[-1]`;

`=> t = *(p + 4 - 1)`;

`=> t = *(p + 3)`;

which is nothing but `p[3]`. `p[3]` is the starting address of string "gh" . So output is "gh"

Note : `p` is a pointer to an array of character pointers where each pointer points to the starting address of the corresponding string.

12 ^ | ▾ • Reply • Share ›



shabi victor ➔ Mr. Lazy • a year ago

how r u taking

`sizeof(int) = 4`

it should be 2...

^ | ▾ • Reply • Share ›



pranav adarsh ➔ shabi victor • 3 months ago

you are right but it may be, not should [be.it](#) depends on which compiler your code is running!!!

1 ^ | ▾ • Reply • Share ›



SMRUTI RANJAN SAHOO → shabi victor • 8 months ago

Size of int is 4 byte . Actually it depends upon compiler . So we generally take it 4 byte .

^ | v • Reply • Share ›



Feliz Kumar • 2 years ago

here t contains the base address of the argv[].so t contains address of "ab" and since the array is character array after every 1 byte of storage the strings are stored.Now p=p+sizeof int stores the address of ij in p since it moves to the four places after ab since it is sizeof(int)=4 and it is storing is character of 1 **byte.so** p[-1] gives to reduce the pointer to 1 and results in 1 less than the address pointed to ij and address is **gh.so** *t gives the content at address gh ie "gh".

5 ^ | v • Reply • Share ›



Alex Burns → Feliz Kumar • 2 years ago

You're right, but we shouldn't be assuming that sizeof(int)=4, as this is machine dependent. This should be explicitly stated at the beginning.

1 ^ | v • Reply • Share ›



Rishi Bhardwaj • 2 years ago

pls explain this

3 ^ | v • Reply • Share ›



Jon • 3 years ago

So how does this work? No explanation or discussion

3 ^ | v • Reply • Share ›



chillchirag • 2 years ago

The answer can be cd also if the size of 'int' is 2. I ran it on turbo giving me answer as cd.

1 ^ | v • Reply • Share ›



Alex Burns → chillchirag • 2 years ago

Correct, you shouldn't assume sizeof(int) is 4. It's true on modern intel but not on every hardware.

1 ^ | v • Reply • Share ›



Rashid Khan • 2 years ago

pl explain

^ | v • Reply • Share ›

Subscribe Add Disqus to your site Add Disqus Add Privacy

DISQUS

@geeksforgeeks Some rights reserved

[Contact Us!](#)

[About Us!](#)