

# Hack postgres Source Code: Vol I

---

## Chapter 1 : C & Rust

---

### 1.33 Function Pointers - C

Function pointer in C has capability to hold the address of a function.

```
#include <stdio.h>

int add(int a, int b);

int main() {

    // Declare function pointer to take address of `add` function.
    int (*ptr)(int, int) = &add;

    // Dereference function pointer and pass input parameters
    int result = (*ptr)(10,20);

    // We can also call function pointer like
    // int result = ptr(10,20);
    // In this case, we need to initialize function pointer
    // without `&` like int (*ptr)(int, int) = add;

    printf("%d\n", result);

}

int add(int a, int b) {

    return (a+b);

}
```

`int (*ptr)(int, int)` means `ptr` is pointer to a function that takes two integer types (`int`, `int`) as input parameters and returns integer type `int`.

`int *ptr[10]` means array of 10 integer pointers. `int (*ptr)[10]` means `ptr` is a pointer to an array that holds 10 integers.

So, having precedence info in mind about operators kind of important while dealing with pointers in C.