POINTER

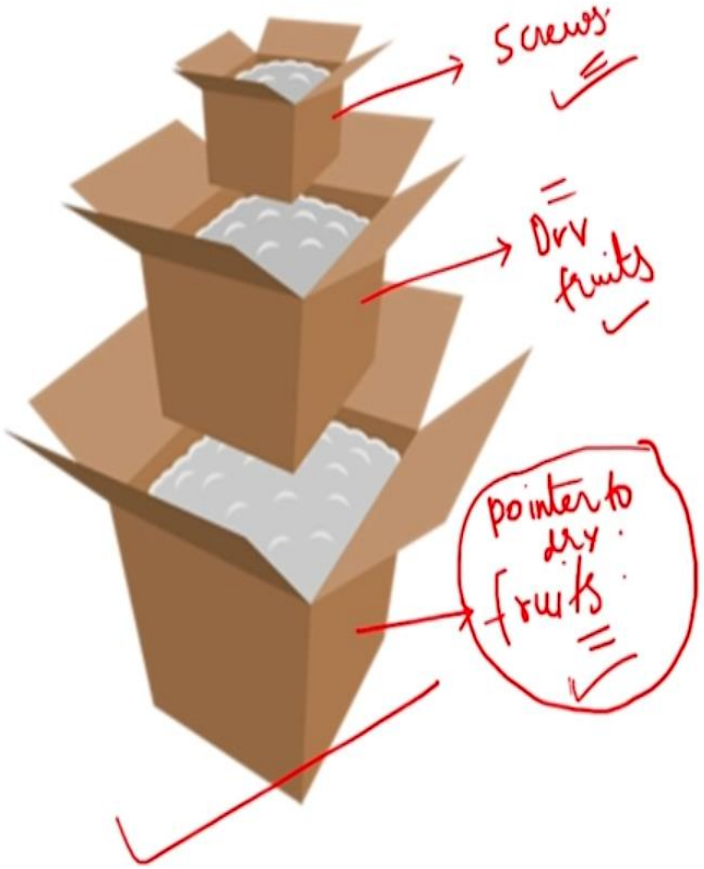
(CODE WITH HARRY)

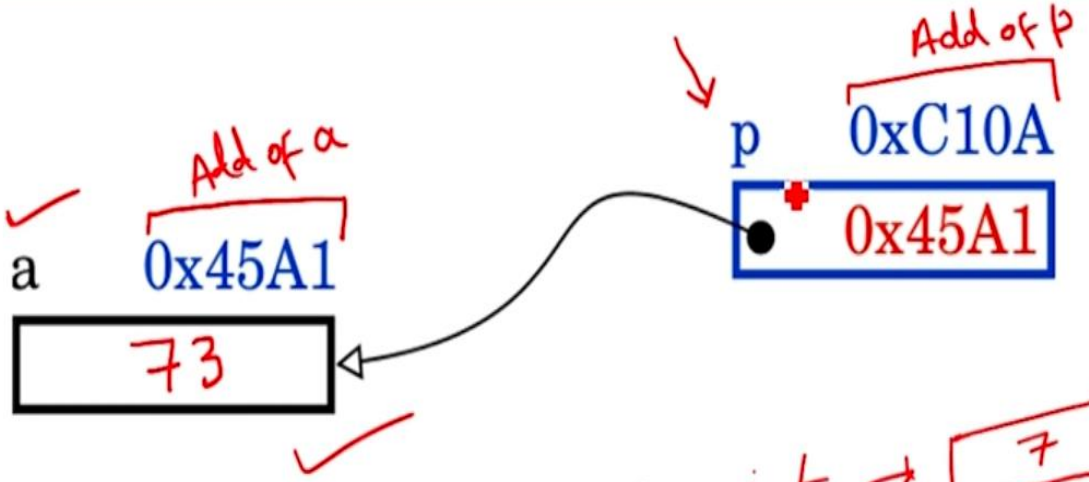
WHAT IS A POINTER?

int float Char

- Variable which stores the address of another variable.
- Can be of type int, char, array, function, or any other pointer.
- Size depends on the architecture. Ex 2 bytes for 32 bit
- Pointer in C programming language can be declared using * (asterisk symbol).

INTUITIVE ANALOGY – STACKED BOXES

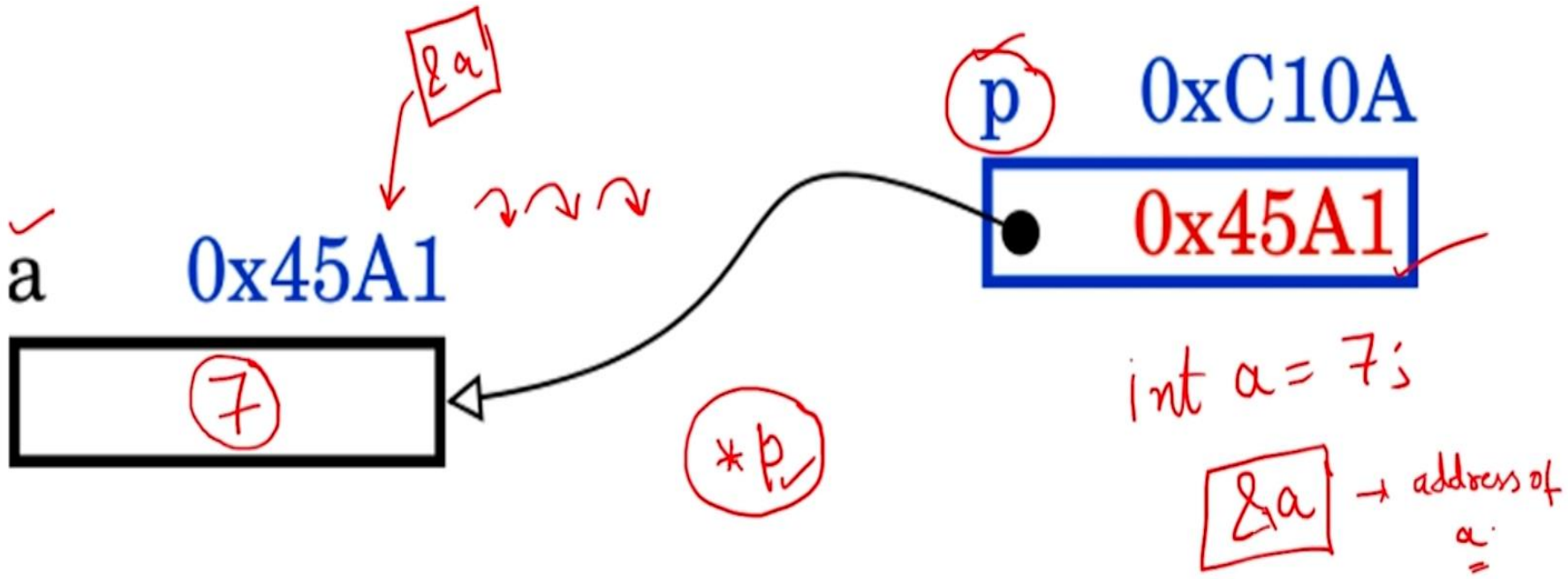




- a is an integer variable
- p is a pointer to integer

int *ptr = &Var;

'&' AND '*' OPERATORS



- The address of operator '&' returns the address of a variable
- * is the dereference operator (also called **indirection operator**) used to get the value at a given address

File Edit Selection View Go Debug Terminal Help Tutorial26.c - C Tutorials Course - Visual Studio Code

EXPLORER

OPEN EDITORS

C TUTORIALS COURSE

Tutorial10.c
Tutorial10.exe
Tutorial11.c
Tutorial11.exe
Tutorial13.c
Tutorial13.exe
Tutorial14.c
Tutorial14.exe
Tutorial16.c
Tutorial16.exe
Tutorial17.c
Tutorial17.exe
Tutorial18.c
Tutorial18.exe
Tutorial19.c
Tutorial19.exe
Tutorial20.c
Tutorial20.exe
Tutorial21.c
Tutorial21.exe
Tutorial22.c
Tutorial23.c
Tutorial23.exe
Tutorial24.c
Tutorial24.exe
Tutorial26.c

OUTLINE

Tutorial22.c | Tutorial23.c | Tutorial24.c | main.c | Tutorial 25.c | Tutorial26.c | Settings | Interactive Playground

```c
#include <stdio.h>

int main()
{
    printf("Lets learn about pointers\n");
    int a=76;
    int *ptra = &a;
    printf("The address of pointer to a is %p\n", &ptra);
    printf("The address of a is %p\n", &a);
    printf("The address of a is %p\n", ptra);
    printf("The value of a is %d\n", *ptra);
    printf("The value of a is %d\n", a);
    return 0;
}
```

0 0 main() Go Live Ln 10, Col 44 (4 selected) Spaces: 4 UTF-8 CRLF C Win32

Type here to search

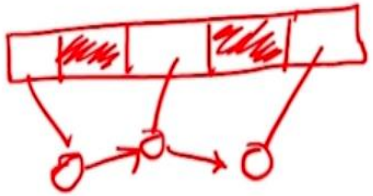
Subscribe

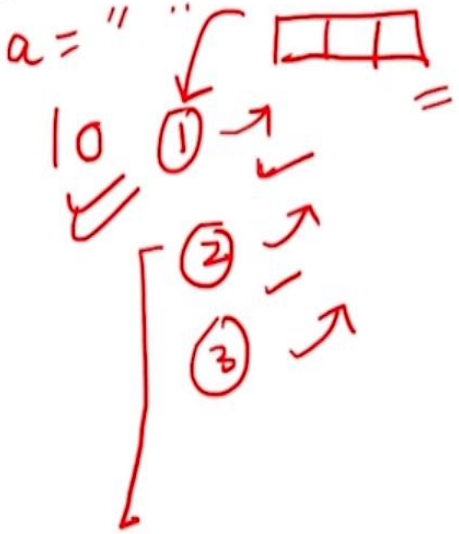
NULL POINTER

- A pointer that is not assigned any value but NULL is known as the NULL pointer.
- In computer programming, a null pointer is a pointer that does not point to any object or function.
- We can use it to initialize a pointer variable when that pointer variable isn't assigned any valid memory address yet.
- int * ptr = NULL;

USES OF POINTER

- Dynamic memory allocation
- Arrays, Functions, and Structures
- Return multiple values from a function
- Pointer reduces the code and improves the performance





ACTUAL AND FORMAL PARAMETERS

- When a function is *called*, the values (expressions) that are passed in the call are called the *arguments* or *actual parameters*. [x, y]
- *Formal parameters* are local variables which are assigned values from the arguments when the function is called.

```
int add (int a, int b) {
    int y1; // local variable
    return a+b;
}
```

```
int z; → global var.
int main() {
    int x=2, y=3;
    int s = add(x, y);
    x=y, → error.
}
```

CALL BY VALUE

- When we call a function by value, it means that we are passing the values of the arguments which are copied into the formal parameters of the function.
- Which means that the original values remain unchanged and only the parameters inside the function changes.

CALL BY REFERENCE

- The **call by reference** method of passing arguments to a C function copies the address of the arguments into the formal parameters
- Addresses of the actual arguments are copied and then assigned to the corresponding formal arguments