Pointers: Dynamic Memory Allocation

Used to save space

Pointers

* Hold addr of other pointers, make memory smaller save memory
* If x = variable, &x return x address.
* Statement void \* declares the pointer to any type.
* p1 = &x; The pointer p1 “points to” x
* legal: int \*p, q;

Outputs

X = 4;

P1 = &x;

Printf(“%d/n”, \*P1);

OUTPUT = 4

(\*p1) = 7;

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

OUTPUT = 7

(\*p1)++;

Printf(“%d\n”,x)

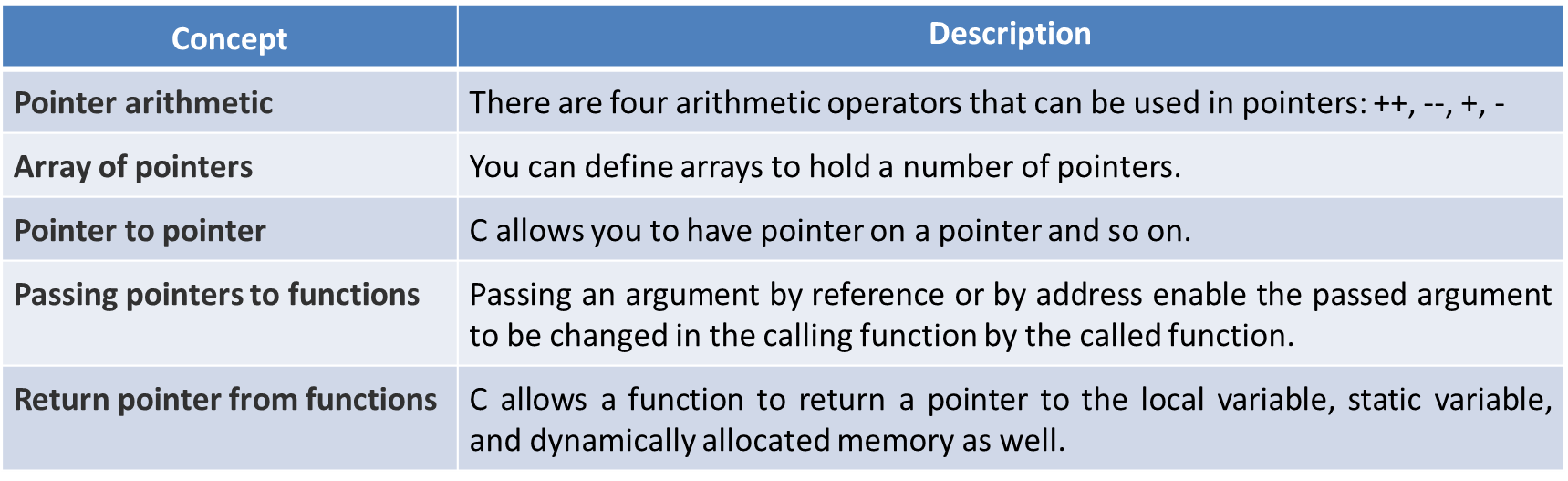
OUTPUT : 8

y = (\*p1) + 5;

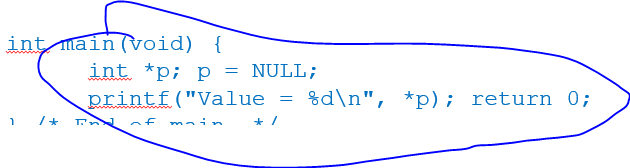
printf("%d\n", y);

OUTPUT :13

Pointer Concepts



Segementation Fault



Here what happens is the pointer p got dereferenced (a.k.a no space was made for it) so it couldn’t print because it basically doesn’t exist.

**Dynamic Storage Allocation**

Recall Data Structures

GET

**Linked list in C**

**GET**