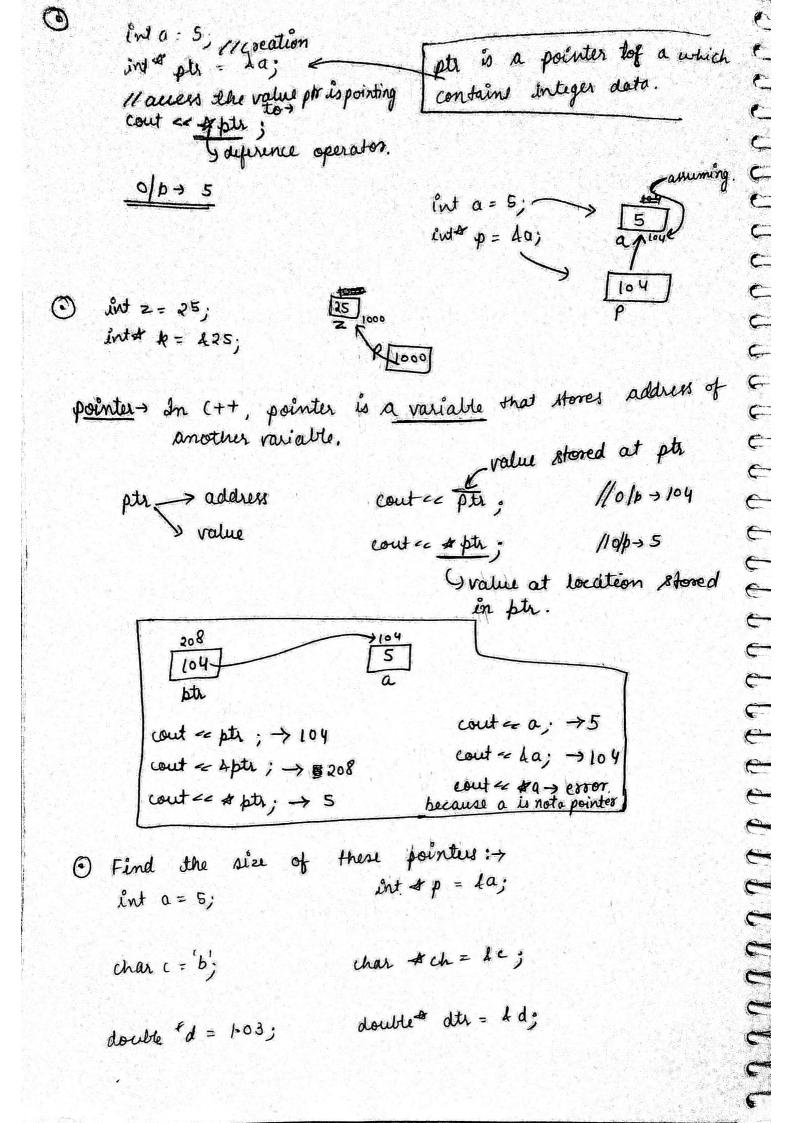
int a = 5; In minory, 5 at integer block stored 5 in But we can't give name to a mem low. The only way of accessing is to use addition. Symbol Table is to use addition. In aymbol table variables are mapped with their addresses. Dees a = 5, b = 5 points to same memory block? int a = 5; int b = 5. We don't decide the addresses of these variables. This memory mynt is done by OS. Int a = 5; cout = a; Int a = 5; cout = a; Int b = 5; cout = 2 (10/p) 5 Low by address of (2) operator. int b = 5; cout = 2 (10/p) 5 Low by address of a variable? This address of a variable. pointer > 4 type of databypt that store address.	Poenters - 1
with their address. Does a=5, b=5 points to same memory block? int a=5; int b=5. We don't decide the addresses of these variables. This memory mgnt is done by 05. Don we find out the address of a variable! They address of (2) operator. int a=5; cout = a; Int b=5; cout = 1a; o/b > 0xffabcdef1234 This address of be will be different from address of a. pointer > A type of detatype that store address. is a int a=5. is a int a=5. is a int a=5. is a int a=5. cout = 2 da; pointer > A type of detatype that store address. is a int a=5. content of a. pointer > A type of detatype that store address. is a int a=5. content of a perator. detatype of syrtax variable integer data. detatype of syrtax variable address.	ant a = 5; In memozy, 5 a letteger block stored 5 in But we can't give name to a mem loc. The only way of accessing
int a = 5; int b = 5. We don't decide the address of these variables. This memory mgnt is done by 05. O han we find out the address of a variable? I'ver, by address of (2) operator. int a = 5; cout = 2 da; Int b = 5; cout = 2 la; o(b > 0xfrabcdef1234 there address of a different from address of a coutre of a. pointer > 4 type of distant pt that store address. into a int a = 5; integer lint of p = la; revisable destart of a coutre data. destart = 2 data. destart = 2 data. pointer to address.	with their addresses.
int a=5; cout = a; loop > 5 Lout = la; lot b=5; cout = ib; o/b > 0xfrabcdef1234 This is the address of int b=5; cout = ib; o/b > 0x45321 This address of b will be different from address of a. pointer > A type of detatypt their store address. is a int a = 5. sointer of integer lint of p = la; > variable integer deta. detatype //syrtax variable carme.	int $a = 5$; $[004]$ $[008]$ Symbol Table $[a]$ int $b = 5$. $[5]$ $[5]$ $[a]$
Lout = 2 da; //0/p > 5 Lout = 2 da; 0/p > 0xfrabcdef1234 This is the address of int b=5; cout = 2 lb; 0/p > 0x45321 — This address of b will be different from address of a. Pointer > 4 type of datatype that store address. is a int a = 5. so integer int by p = 4 a; rewiable integer data. Catalype //syrtax variable careference name.	O lan we find out the address of a variable! ⇒ Yes, by address of (2) operator.
be different from address of a. Pointer > A type of destatype that store address. is a int a = 5: pointer to integer (int to p = 4 a; > variable data. > address of operator. destatype / syrtax variable name.	
is a int a = 5; pointer to pointer to integer lint to p = 4 a; vouiable integer data. datatype //systax variable dereference name.	be different from address
datatype L/systax variable dereference name.	is a int a = 5.
	deseference name.

pointer to the data. The ptr = lch; sptr is a pointer to their data.



All three pointers size will be same, because pointer stores address of variables all all and address size of 7 any type of data will be same. (no matter these datatype is cout = size of (p) = size of (ch) = size of (dtr); ->0/10 - 888 cout a size of (a) = c six of (c) = c six of (d); 0/63 4 18 Here the size of pointer is &. Architecture dependent. Find out? @ what do you mean by 64 system? 1 Why pointer's size is · Why do we need pointers? Synamic memory allocation 4 Memory mgmt → To access hardware. > Pointers authoretic, handling NULL pointers >> To pass a function as an argument inside another function. O seclaring a pointer > ent to ptr; // declaration. In this case a garbage value is assigned to the ptr. cout = *ptr; and here we are trying to access that memory. We get <u>segmentation</u> fault. (meuns we brying to access that memory which either don't enist or out of the alloted memory space tof program.

This is a <u>BAD PRACTISE</u>. Now how to correct this?

6 Null pointer.

Int to ple = 0; tun is how rule pointer is

```
means that pointer is pointing to nothing.
Error - segmentation fault.
 But here we can check this before kunning
       if (ptr = = 0)
                                                            cout « " ptr is a NULL pointer ",
 Other way to weate NULL pointer >
        ento ptr = NULL;
 New/modern way to create NULL pointer >
         int # ptr = nullptr;
 → int a=5;
    lut of ptr = La;
    a = a+1;
                                ptr +1 = 108
     dr = pht 1;
                                                           er int a=5;
    into p = la;
    Ab = $6+1 000
                       $ b = 5
   value at
 address stored in p
                       4 p = # p+1
                             5+1=6
                         a JUH 12 18 36 15
             208
         por [104/
                                -> ++ (4phr) -> 12
                               → a= a+1 -> 11+1=13
     79 -10
     -> 4a -> 104
                               → pp= *pf+2 → # 15
     -> pr -> loy
                               -> # ptr= & ptox 2 -> 30
     -) Aptr -> 10
     -> dptr-> 208
                               -> Apro=(4 11/2) -> 15
     > #ph*2 -> 20
     -> (Aptr)++ -> 10
```

```
⇒© (an a pointer copy another pointer).
             Yes,
                             - p is also as pointer.
-3
3
      ent a = 5;
      into p = a;
73
                       - gives error so court assign pointe =
                           a with a value, we have to give address.
3
       int a = 5;
3
       intop = 4a;
7
       But # dpts = ptr;
                         -> Lopy a pointer into another.
3
                             STIR.
                                          4p - 5
3
                            1041
                                           $dpts -> 5
1
3
    a \rightarrow 5
                     Q -> 104
    La 104
9
                     22 7312
    P-> 104
                     #q > 5
3
                    ($p/2) -> 2 -> 5/2 = 2
    AP -> 208
9
   4\rho \rightarrow 5
                               -> 5/2 ->2
                     89/2 × 2
J
                                                              2000
                                                        1000
      int a = 10;
      Puttp = da;
      int & q = p;
      contra e la expertence *peque la cela extre
          Tec 47 ec 88 ez (8p+89+48) ez (4p)*2+(40)8
       2c (4p/2) - (8q/2) er endl;
                               19 -1000
    a+ 10
                               49 -10
    da - 104
                               r -> 104
    P - 104
                               Ar -> 2000
    Ap > 2160
                               87 - 10
                              (#p+#q+#r) -> 10+10+10=30
   Ap + 10
                              (#p) * 2 + (xr) +3 -> 10x2 +10x3 = 50
   Q -> 104
```

(#P/2) - (#9/2) -> 10/2 - 10/2 = 0