**POINTERS**

It stores the address

Dclared

Dt \*ptrName ;

**Note Pointer doesn’t belong to any kind of data type ; it jst hold the address the address is any data rype**

Ptr is a spl variable which is pointing to integer data type –(R-L) == int \*ptr;

Null pointer – int\*ptr = null;--null pointer

Void pointer—generic pointer(it can point any type of datatype)

Wildpointer –with out any data specialized to the pointer then it is wild pointer

Dagling pointer(it is a situation)

…. The adrees may have 4/8 for any data type in pointer…….

A computer screen shot of a code

Description automatically generated

Generally pointer give address of that value but if we want the value we \*ptr (de referring takes place)

**Errors that occur 🡪 invalid use of void expression –(generic pointer)**

So we want to do type conversion 🡪 \*(int \*)ptr

In general pointers --- if we initialize to int but we want it in float we can do it by using type conversion ant execute state

Pointer can notated as array in -- \*(ptr+0) to get arr[0]

**THUMB RULES**

1. **&\* = nullify each other (no symbols)**
2. **Op[] => \*op**
3. **\*op => op[]**
4. A screenshot of a computer

   Description automatically generated**\*op(i+0)**

Dynamic address:

If we want get unnamed address then we have to create memory allocation

Here we malloc calloc –stdlib

Malloc 🡪only allocation of memory …it return void , \* ---

The below example show how memory allocated to ptr

**Synrtax --- void malloc(size ) calloc used 2inputs**

But here

A screenshot of a computer

Description automatically generated

If(ptr==NULL);

{

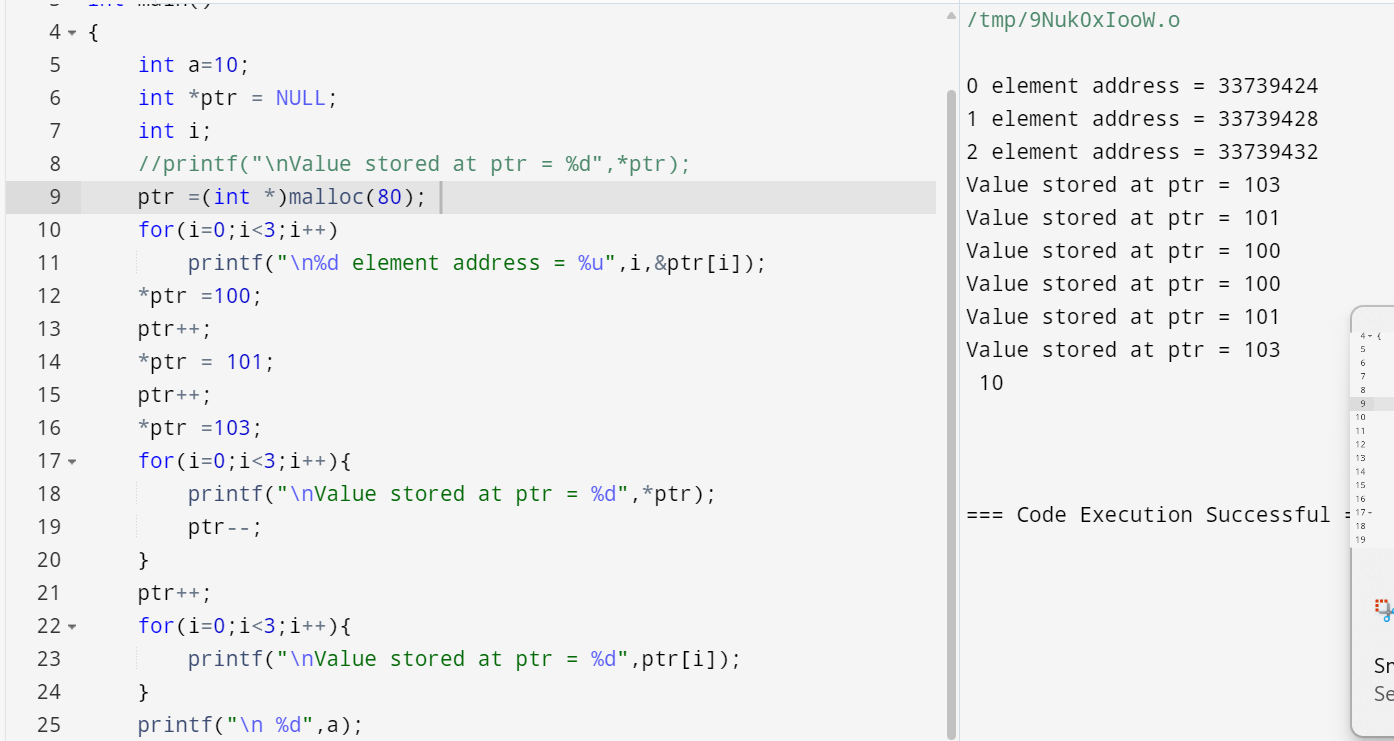
Perror(”malloc:”);(perror is used when we use stdlib)

Exit(0);

}

Note:

1. The malloc only store what the memory we assigned
2. If there is more than the memory we assigned also it can take that but it stores it temporarily we don’t know whether it may be or not
3. If it is larger memory then we know that it is out space so to get that output just we through error so for that we (perror)



Note : remember in pointers we should not losss the base address becoz we can get the total o/p from start

so it best activity to store the values in temp memory location so that we retrieve it

wapr to find value if it present in the list

Daggling : **pointer is pointing the address where it has been the address has been destroy**

Storage **classes: static , register,auto**

**Static**

By using static key word we can arrange the memory allocation if there was no memlory but it will be until the process remains

A screenshot of a computer

Description automatically generated

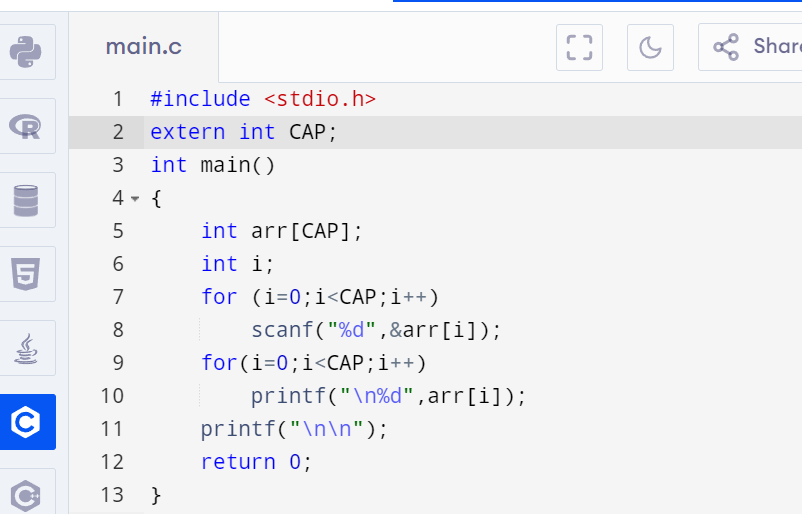
STATIC key word it scope is until the process ecxits (a is global variable so the value will be updated and change) it cant access directly but the scope of variable remains and once it declare it will not be declare but it updated value is just used )

A screenshot of a computer

Description automatically generated

**REGISTER** : by using register key word the data or memory is allocated in cpu so the to access the variable is faster because it stores in cpu some variables that which we use frequently is used

**MUTABLE ARRAYS : the declaration that we done im another file but same tool it can access but it can be done by only EXTERN KEY word**

****

**FUNCTION POINTER ;**

**Syntax *; type(\*fptr)();***

**A screenshot of a computer

Description automatically generated**

**While we malloc and calloc we have to free the space:**

While doing this what are the errors we may occur

Free(); double free detected intachahe 2

Aborted (core dumped)