**Day09**

**Arrays**

/\*initilization

decalaration part -->

syntax: data\_type arrname[size];

ex: int arr[10];

ex: float arr[10];

ex: struct{

int id:

char name[20];

};

struct emp e[10];

\*/

//access elements of arr to store or evealute

/\*arrname[indexvalues]=valu;

indexvalue from 0 to size-1

arr[2]=20;

how array internaly work

baseaddress+indexvalue\*sizeofdatatype --->formula

for example base address is 2002

arr[2]={1,2}

arr[0]=1

arr[1]=2

2002+0\*2

2002

2002+1\*2

2004

1.static array --->fixed sixe

2.dynamic array -->heat memory

3.stretchable array

4.mutable array

1.static array: the size of array is known before to the complietime

ex:

int arr[5];

2.dunamic array: size of the array is allocated/known at the run time

ex: malloc, calloc, realoc --->stdlib.h

3.stretchable array: size of the array increased or desc dependance on the need for dynamic array

ex: malloc, calloc, realoc

4.mutable array: the size of the array is known or allocated at the time of linking and before execution

Initilization

Int a[5]={1,2,3,4,5};

Int a[]={1,2,3,4};

Int a[5]={1,2,3}

Int a[20];

**/\*store odd values in array\*/**

#include <stdio.h>

int main() {

// Write C code here

int n,m,arr[100],i,count=0;

scanf("%d %d",&n,&m);

for(i=n;i<=m;i++)

{

if(i%2!=0)

{

arr[count]=i;

}

}

for(i=0;i<=sizeof(arr[count]);i++)

{

printf("%d ",arr[i]);

}

return 0;

}

**Function**

1.std lib function

Printf, sqrt, abs, pow

2.user defined

User is defining his/her own task to be performed

Syntax : fun\_name(int arg)

{

Statement;

Return rdt;

}

Header files is c

Propotype files is h

#ifndef

#define

/statement protection

#endif

**/\*To store elements in array\*/**

#include <stdio.h>

int main() {

// Write C code here

int i,arr[10];

for(i=0;i<=5;i++)

{

printf("%d",(arr[i]=i));

}

return 0;

}

**/\* store and print \*/**

#include <stdio.h>

int main() {

// Write C code here

int i,arr[10];

for(i=0;i<=5;i++)

{

scanf("%d",&arr[i]);

}

for(i=0;i<=5;i++)

{

printf("%d",(arr[i]=i));

}

return 0;

}

/\***find largest element in an array\*/**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

