\*arrays:

it occupies same space as individual elements,works faster to store and retrive elements,logic writing is easy,ifyou know the base address of array we can iterate throughtout the array ,individual variables stored in random locations in memory where as since array is a sequence of bytes it will store them in adjacent locations ,it is the sequence of bytes

1st cell of array is called as base address (s[0]),by default every array name codes it base address

-->drawbacks of array:

\*different type of elements cannot be stored

\*once the size is defined we cannot alter the size,that's why an array is known for its static memory allocation

\*wastage of memory

\*s[name]--implicit array for this we need to use dynamic memory allocation

\*returns garbage value for the leftover cells if the values are entered during runtime,returns zero if the values are entered in the program itself

\*if you are trying to insert an element in the middle of the array we have to move all the existing elements to the next cells manually

\*when you delete an element from an array again we need to write logic to move the next elements to previous indices.

\*types of errors-syntax error,logical error,runtime error

exception is something which can be handled,error cannot be handled.

\*sometimes memory may not be sufficient to store all elements of array if size is too big

\*two dim array--int a[50][3];

50 students 3 subjects marks(m,p,c);

int a[20][6];

20 overs 6 balls

\*three dim array::

int s[4][2][10]

4 years

2 sems

10 subs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*functions:

group of statements,which returns a value

types of functions

-user defined(eg:main())

-built-in(system defined/pre defined)

---entry point of c program is main()

---#include<stdio.h>--preproccessor directive

-- we can pass arguments to main()function which is called command line arguments

eg: int main(int argc,char\*argv[])

--we can read values without using scanf by using command line argument

--default return type of a function in C --int

-- empty main and int main are same (returns integer)

to create a user defined function:

----->1.proto typing(declaration)

2.definition(body)

3.function call

-->we can write many funcs without main() and compile it and we can also call the funcs in an another program

#include<stdio.h>

void sum(int x,int y)

{

printf("sum:%d",x+y);--we should only compile this we cannot execute it

}save it as vaishu.c

now we can use it in an other program to execute

#include<stdio.h>

#include "vaishu.c"

{

main()

{

sum(100,200);

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*pointers

-->pointer is a variable which stores the address of another variable that belongs to the same data type.

-->void pointer is used to store the address of any type of variable,it is a keyword as well as data type,this is also know as generic pointers.

-->pointer size is always static type,i.e,it has same size for all the data types.

--> the execution is faster as we can access the memory

-->dynamic memory allocation

-->efficiency of the program will be increased

-->call by reference

-->drawbacks:

these are not secured because we share addresses.

\*\* storage classes in pointers---

auto-garbage value

static-0

extern-0

register-garbage value

\*structures

-->sizeof-keyword

-->exit-function,is from the header file called stdlib

145=1!+4!+5!=1+24+120=145--strong number

adding prime asci values\*\*

p=&s[0]; (or) p=s;

\*pointer gets increased according to its size(data type)eg: for(i=0;i<10;i++;p++)

printf("%d",\*p);

\*\*sorting:

arranging the data in ascending or descending order.sorting arranges data in a sequence which makes searching easier.

different types of sorting techniques:

\*bubble sort

\*selection sort

\*insertion sort

---->selection sort:

this algorithm will first find the smallest element in the array and swap it with the element in the first position,then it will find the second smallest element and swap it with the element in the second position,and it will keep on doing this until the entire array is sorted.

-->it is called selection sort because it repeatedly selects the next-smallest element and swaps into the right place.

1)main()-- is a special function whose signature is given by the language and definition given to user choice

2)Signature

<return type>name of fun(parameters)//signature

<return>abc()

3)function: should have a declaration,definition,invokation

4)default return value of any function is integer

<return>name(parameters)

return name(passing parameter)---1

no return name(passing parameters)--2

return name(no parameters)

no return name(no parameter)