group => using structure

int a;
char arr[20];
double c

Syntax => struct struct_name
{
data member1
data member2
data member3
};

ex => struct Student
{
int rollno;
char name[50];
float marks
};

```
struct Student
{
    int rollno;
};
void main()
{
    struct Student ob;
    ob.rollno = 4;
}
```

```
void add
{
}
main()
{
    add()
}
```

inside main

```
struct Student
{
    char name[50];
    int age;
    int rollno;
};
```

```
Void main()
{
    datatype
    struct Student S;
    S.name = "Anop";
    S.age = 25;
    S.rollno = 4;
}
```

#include<string.h>

↓

struct String

{

len

}

Pointer

struct Student{

int rollno

};

```
void main()
{
struct Student S1;
S1.rollno=8;
struct Student *ptr;
*ptr=&S1;
printf("%d",ptr->rollno);
```

Variable

data type

struct add{

int a;

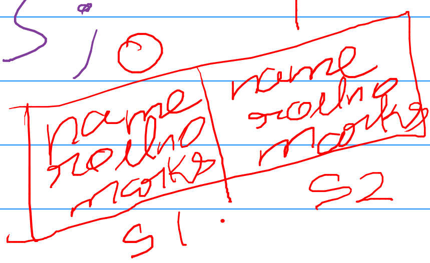
int b;

int c;

};

```
void main()
{
struct add a4;
a4.a=4;
a4.b=4;
a4.c=a4.a+a4.b
printf("%d",a4.c);
}
```

```
struct Student
{
  int rollno;
  int marks;
  char name[ ];
}; 10
```




```
void main()
{
  struct Student S[2];
  S[0].rollno = 4;
  S[0].marks = 84;
  S[0].name = "Anup";
}
```

File handling → Write ✓
Create ✓
read ✓
modify ✓

↓

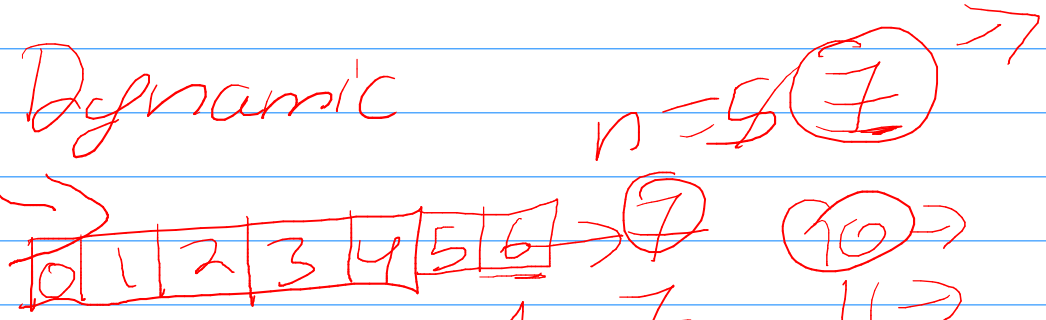
file stored in a data

	Function	Purpose
1)	fopen()	open file
2)	fclose()	Closing the file
3)	fprintf()	Write formated data
4)	fscanf()	Read formated data
5)	fgets() / fputs()	Read / write string

6> fread() / fwrite() | Binary Read/write

Mode	Meaning
"w"	Write (new file or overwrite)
"r"	Read only
"a"	append to end of file
"w+"	Read & Write

(7) Static
Dynamic



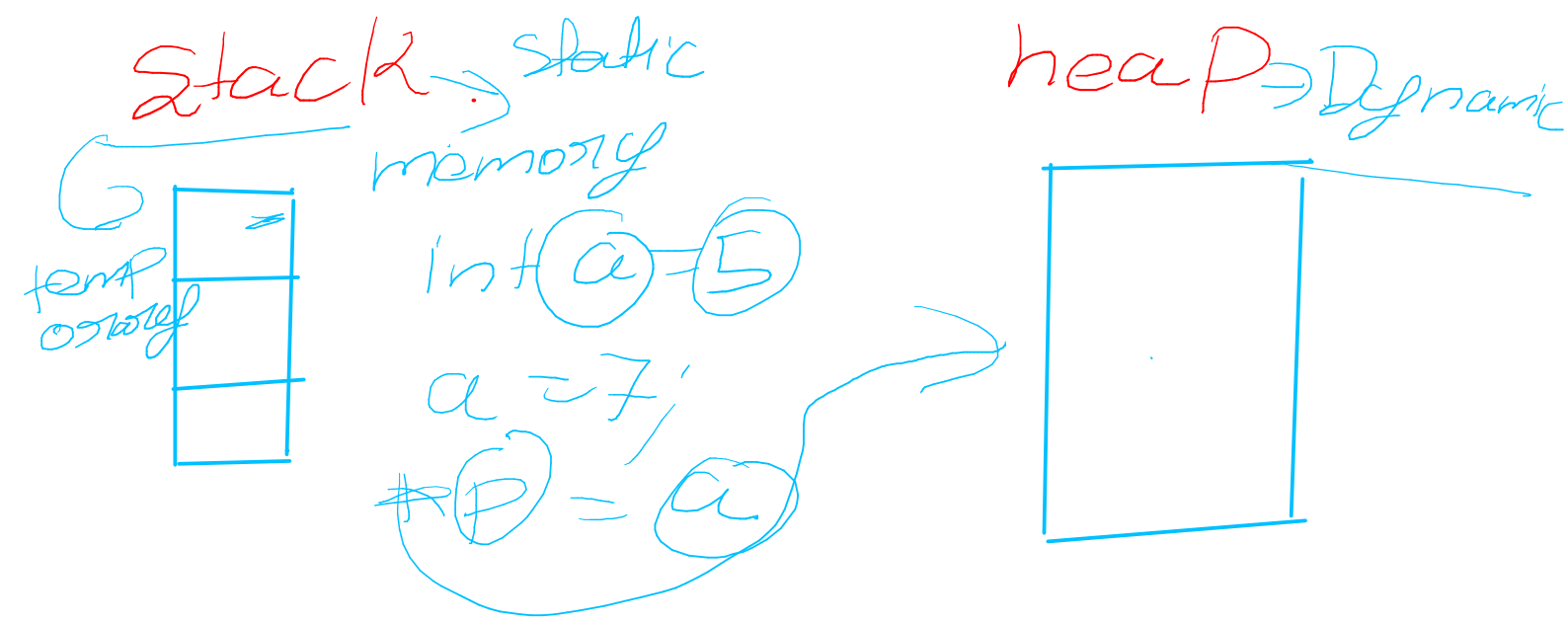
Compile time Runtime

"Hello" → compiler → 1010101

```
#include<stdio.h>
  Void main()
  {
    int a=4, b=0;
    int c=a/b;
     printf("%d", &c);

  }
```

Function	Purpose
calloc()	Allocate and zero initilization
malloc()	Allocate memory
realloc()	Resize memory block
free()	(delete) Free allocated memory


Stack → static memory
heap → Dynamic
temp
int a = 5;
a = 7;
*p = a;