

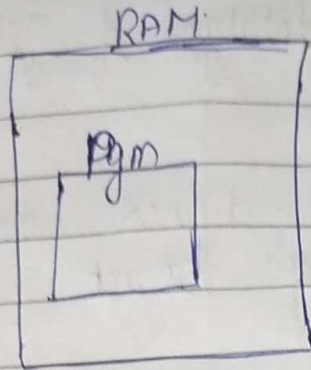
C-120 \Rightarrow File Handling in C - Part 2

File Pointer and fopen() function

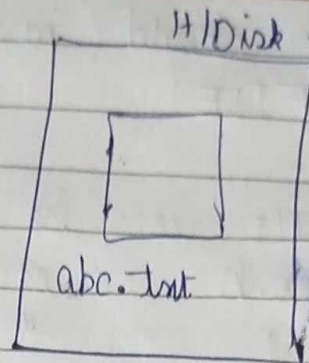
* Whenever we run the program, it gets stored in a main memory (RAM) and once the execution is terminated the stored contents are vanished in RAM.

* When we want to store the program contents permanently then we use hard disk memory.

volatile memory



No



* our data in hard disk is stored permanently using file.

* File is our memory block (or) sequence of bytes where our data is stored permanently.

* We can create, open, read, write and append a file.

* Suppose, we have a file named abc.txt and if we want to access this file we can't simply access it; at first this file should be loaded into the main memory while processing and then to access the file we need a file pointer.

* We know pointer is a variable which stores address of a other variable.

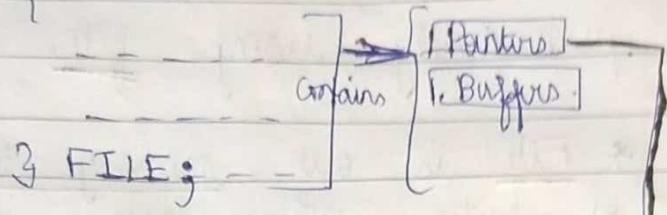
↳ e.g. `int *ptr; int a;` (stores address of variable whose datatype is integer)
`ptr = &a;`

* Likewise the pointer which points to the

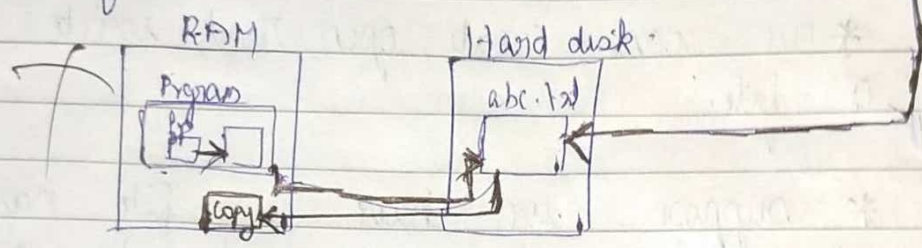
file should have a datatype of file
(FILE)

* FILE is already defined inside
stdio.h using typedef struct

↳ e.g., typedef struct
{

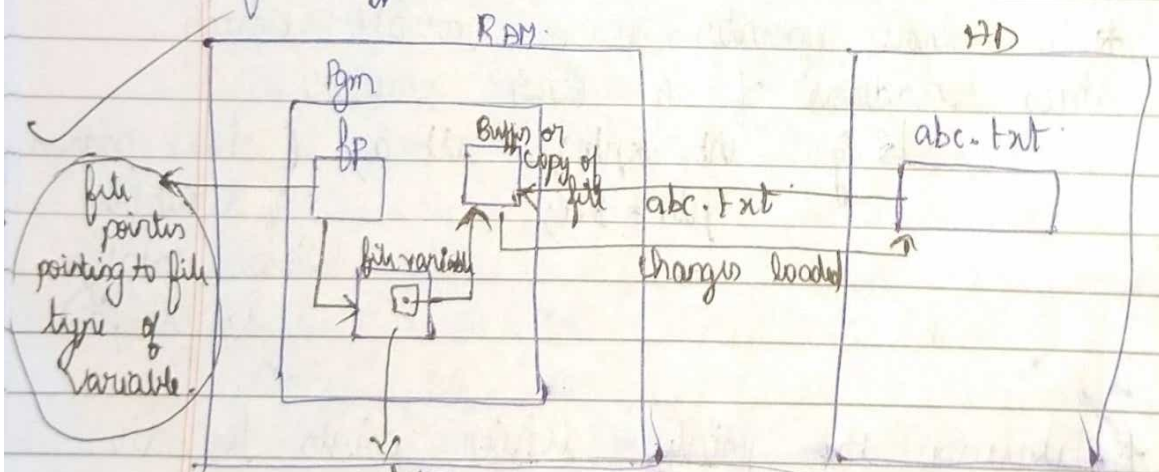


* This pointer is now going to point to
our file



FILE *fp;

* This pointer will have address of a
file type



whatever we
open, read this file all will be reflected in buffer and
loaded into a file in hard disk.

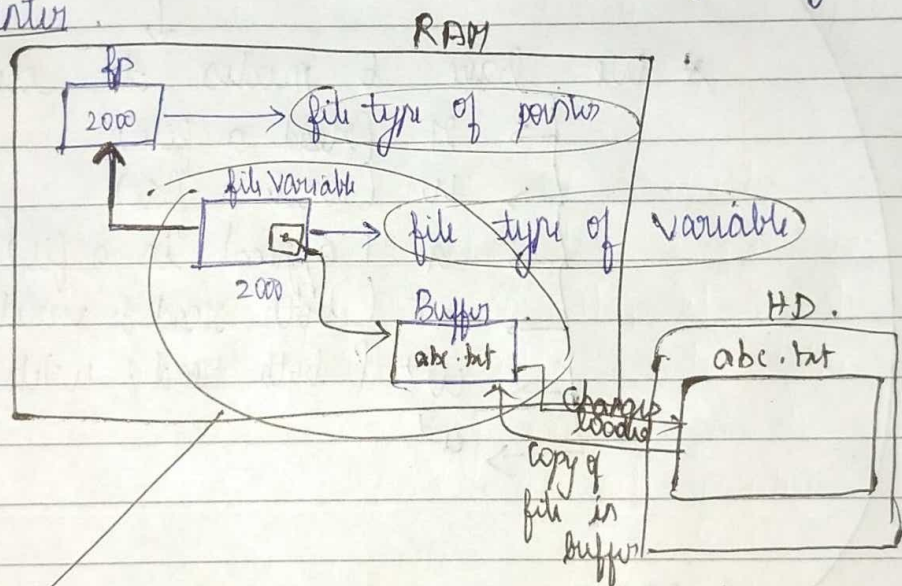
Date _____
Page _____

* For pointer variable `*fp` of type `FILE` datatype we have in our program but for file variable we don't have created in our program.

* How to create a file variable type?

→ For this we have `fopen()` function which is already defined inside `stdio.h`.

→ Whenever we call `fopen()` function, this will automatically create file type of variable containing address and that address is stored in file type of pointer.



→ These two tasks (i.e. Copy of file in buffer and creating a file type of variable) is done using only single function `fopen()` function.

→ Now what are the arguments we are going to pass in `fopen` function?

`fopen("filename", mode)`

what file is
the name of file
we going to open.

In which mode we
want to open the file.

(i)
Eg: `fopen("abc.txt", "mode")`

* We have 6 modes to discuss

- r (read a file)
- w (write a file)
- a (append is a file)
- r+ (both read & write)
- w+ (both read & write but diff)
- a+

* This will return address of file object
or file variable that would be
stored in our special pointer which
stores address of file type of variable.

FILE *fp; (ii) `fp = fopen("filename", "mode")`

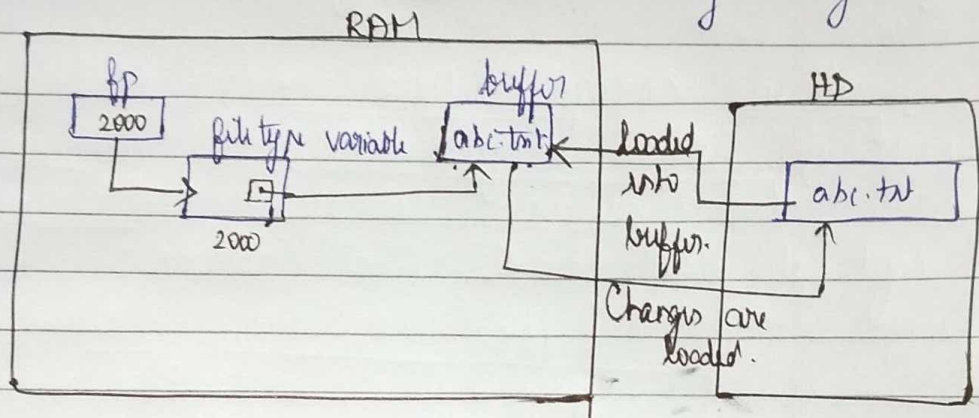
* Now with the help of this fp pointer
we can access the file and do anything with file.

Note (close the file)

* After everything is done, we are supposed to close the file

(3) `fclose(fp);`

→ Once we close this file the memory allocated for buffer and file type of variable will be freed off. (ie) that file can be now accessed by any other



→ Mainly in our file type variable we have a pointer that points to a buffer memory; so that what all changes we do with our file it is reflected in our buffer and finally gets loaded into our permanent storage memory (HD) where our file is located.

Note (we want to read, write in file)

* We have,
`fscanf` → read a file
`fscanf` → write a file
~~fprintf~~ `fputc` → write a character in file.
`fgetc` → read a character in file.
`fgets` → read a string from file
`fputs` → write a string in a file.