

1/5/23

④ Command Prompt: / DOS.
↓ (Terminal)

GUI (Advanced)

: - Commands

① C: | Users | dell >

: - directory
(folder)

② C: | Users | dell > dir

③ C: | Users | dell > cd . resources

cd desktop

④ C: | Users | dell | resource > dir.

Parent folder → Sub folder

⑤ Relative / Absolute path

↓
(Related
to something)

(starting and end point).

⑥ C: | User | dell | resource > dir C: | Users | dell
(Relative).

⑦ cd .. (parent directory)

⑧ .. | (current parent directory)

⑨ C: | Users | city | Desktop | Docs | A | dir C:\X

① Streams:

Pipes & Filter Architecture.

→ Stream Redirection (Tap, Pipe, Tank).

→ `dir > test.txt` (New / Rewritten / Overwritten).

→ `sort` (Display on first digit basis).
(Control Z to stop)

`sort < test.txt` (Read)

`sort < test.txt > sorting.txt` (Read/Write)
(File to File)

→ I/O Redirection, Absolute / Relative Path.

→ For Linux:

`C: | abc |`

(No directory structure)
(only home).

→ For mac:

`ls` (list)

`mkdir` (new directory)

② `cin` } Are objects (structure)
`cout` }
(Keyboard) (Read)
`istream` — `cin`
`ostream` — `cout` }
(Monitor)

③ `if stream`
(input file) (For Reading from harddisk)

④ `of stream` (Header files)
(output file) (Writing).

⑤ `#include <fstream>`

`int main()`

{ `ofstream ofs;`

`ofs.open("text.txt");`

} `ofs.close();`

(For saving
and closing)

⊕ Between open and close we write code.

```
ofs.open("../test.txt")
```

(correctly developed)

for parent folder

```
ofs.open("../testing.docx")
```

:extension not changes data or order
just icon changes.

```
cout << "Hello";
```

(on console)

```
ofs << "Hello";
```

(In file).

⊙ ofs << oct << setw(100) << right << 3243;

⊙ Harddisk minimum size (4KB).

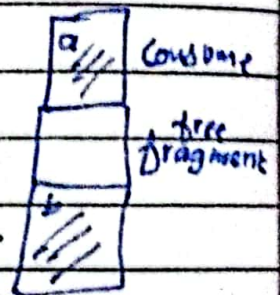
for RAM (1 byte).

⊙ If we write more data now
it would be overwritten.

⊕ For above issue: append

```
ofs.open("test.txt", ios::out);
```

(output mode)



```
ofs.open("test.txt", ios::app);
```

int a=123;

(new data at insertion point)

::app
(append).
(Not to overwrite)

```
ofs << "How are you" << a;
```

⊕ After closing, we can open it again.
after it:

```
ofs.close();
```

```
ofs.open("bad.txt");
```

```
ofs << "OK";
```

```
ofs.close();
```

Above file same, new created.


```
int main()
```

```
{ ifstream fs;
```

```
  if (fs.open("test.txt"));
```

(is-open)
:(File is
present or
not)

```
  if (!fs.is_open())
```

```
    { cout << "File not opened";
```

```
    }
```

```
  char ch;
```

```
  fs >> ch;
```

```
  cout << ch;
```

output('h').

```
  fs >> ch;
```

hellow

```
  cout << ch;
```

'e' (output).

(Ignores leading
whitespaces)

```
  char ch;
```

```
  ch = fs.get();
```

(EOF)

End of file

Character.

(Every compiler
has different
EOF.)

(not ignoring white
spaces).

```
  : for all display : loop
```

```
  while ((ch = fs.get()) != EOF)
```

```
    { cout << ch;
```

```
    }
```

:- stream
not corrupted

because we are
taking in

"char" (It
takes everything)

:- RollNo, CGPA (It stores the data
in file) (Not vanished after
console is closed).

**OBJECT
ORIENTED
PROGRAMMING**
BSE: 2nd
Semester

12/6/23

Quiz - Friday 1:45 - 2:30

*** File Handling:**

[fstream ^{read} - write
ifstream - read
ofstream - write

cin - istream - Keyboard
cout - ostream - monitor.

⊛

(RAM) ← (HD)
(Monitor)

⊛ When extension changes only icon changes
but format not changes.

ofs.open("abc.txt")

ofs << "Hello How are you";

⊛ Char array as unit.

⊛ int a[5] = {10, 20, 30, 40, 50};

for (int i=0; i<5; i=i+1)

{ ofs << a[i] : 1020304050
}

ofs << a[i] << " ";

ofs << a[i] << '\n';

④ OA - '\n' :- hexaValue.

⑤ File must be close at end to be safe from buffer or memory leakage.

ops.close();
⑥ ops.open(.....);

ops.close();

⑦ ops << a[i] << '\n' << flush; :- 3 characters
:- flush
:- endl → '\n' + flush.

⑧ order of matrix not known while reading. For this we should tell the reader that first line represent order of matrix.

2 2
[1 2]
[3 4]

2 1 (order)

[2 1]

⑨ ifs.good(); 1
ifs.fail(); 0

ifs.eof();

⑩ getline(msg, 80)
 ↓ ↓
 char array size

: not ignores
leading
whitespaces.

while(!ifs.eof())

{
 ifs >> a.rows >> a.cols;
 if(ifs.eof())
 break;

④ ofstream always open in output mode.

④ ifstream always open in input mode.

Default: $\left[\begin{array}{l} ifs.open("abc.txt", ios::in); \\ ofs.open("text.txt", ios::out); \\ fs.open("studentDB.txt", ios::in/ios::out); \end{array} \right.$

④ append makes new file if not present otherwise it starting writing from EOF.

```
while (!fs.eof())
{
    fs >> s.rollNo;
    fs.ignore();
    fs.getLine(s.name, 30);
    fs >> CGPA;
    cout << "\n" << s.rollNo << " " << s.name
         << " " << s.CGPA;
}
```

14/6/23

* File Handling:

```
(ifs.get())
a = ifs.get
```

start	after get	
0	1	2 3 4 5
A B C D E		

④ ifs.peek(); If you wanted to get next element while pointers remain at same position.

$a = ifs.peek();$

④ ifs.tellg(); If you wanted to get index of where you are pointing.

`a = ifs.tellg();`

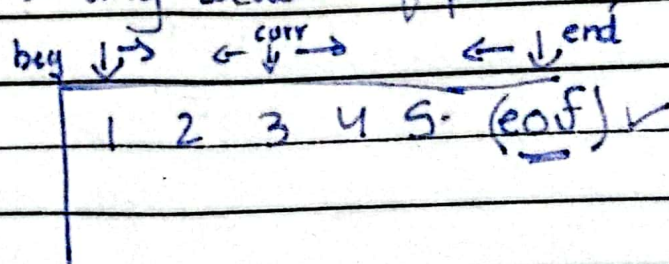
④ `ifs.seekg()`; To move the position of pointer at any location.

`ifs.seekg(-2, ios::end);` (always starts from file)

`ifs.seekg(2, ios::beg);` (always will be positive)

can be positive or negative `ifs.seekg(2, ios::cur);` (from current location to any location of pointer)

from EOF you wanted to check character from start you check character



⑤ `eof()` value is -1.

`int a = eof();` : a = -1

⑥ `'\n'` = `|r|n` (2 bytes)
start of line newLine

⑦ `while (c = ifs.get())`

`{ cout << c; }` :(ABC)

(ABC) eof

`while (ifs >> c)`

`{ cout << c }`

(returns at end the previous store again)

:(ABCC)

while(ifs)

{ cout << c }

(It can also be used).

⑧ `\r` → carriage return
`\n` → line feed

* Stream Status:

foul bit();
 bad bit();
 eof();
 good();

EOF → value

is - 1

Trunc
ale

in
app
out

(current
use)

* For size:

`ifs.seekg(0, ios::end)`

(Gives us size of file)

* `bool FindRecord (fstream &fs, int rollNo)`

{ student s;

while (fs.eof())

{

fs >> s.rollNo;

fs.ignore();

fs.getline(s.name, 30);

fs >> s.CGPA;

if (s.rollNo == rollNo)

return true

ios::app;

* 3.7 → old, 3.75 → new (:\n removes)

(Removing record) (For this issue resize set

10 Sameer 3.7\n

10 sameer 3.75

10
==
20
==

10
sa..
3.7

\n

RollNo
Name
CGPA

* "-1" and "26" are dangerous number

"-1" value of eof();

No value of 26 (No ASCII value)

File handling:

1 2 3 4 5 | -1 6 7 8 9

26
(stops)

④ remove (fileName);

④ rename (old File Name , new File Name);