

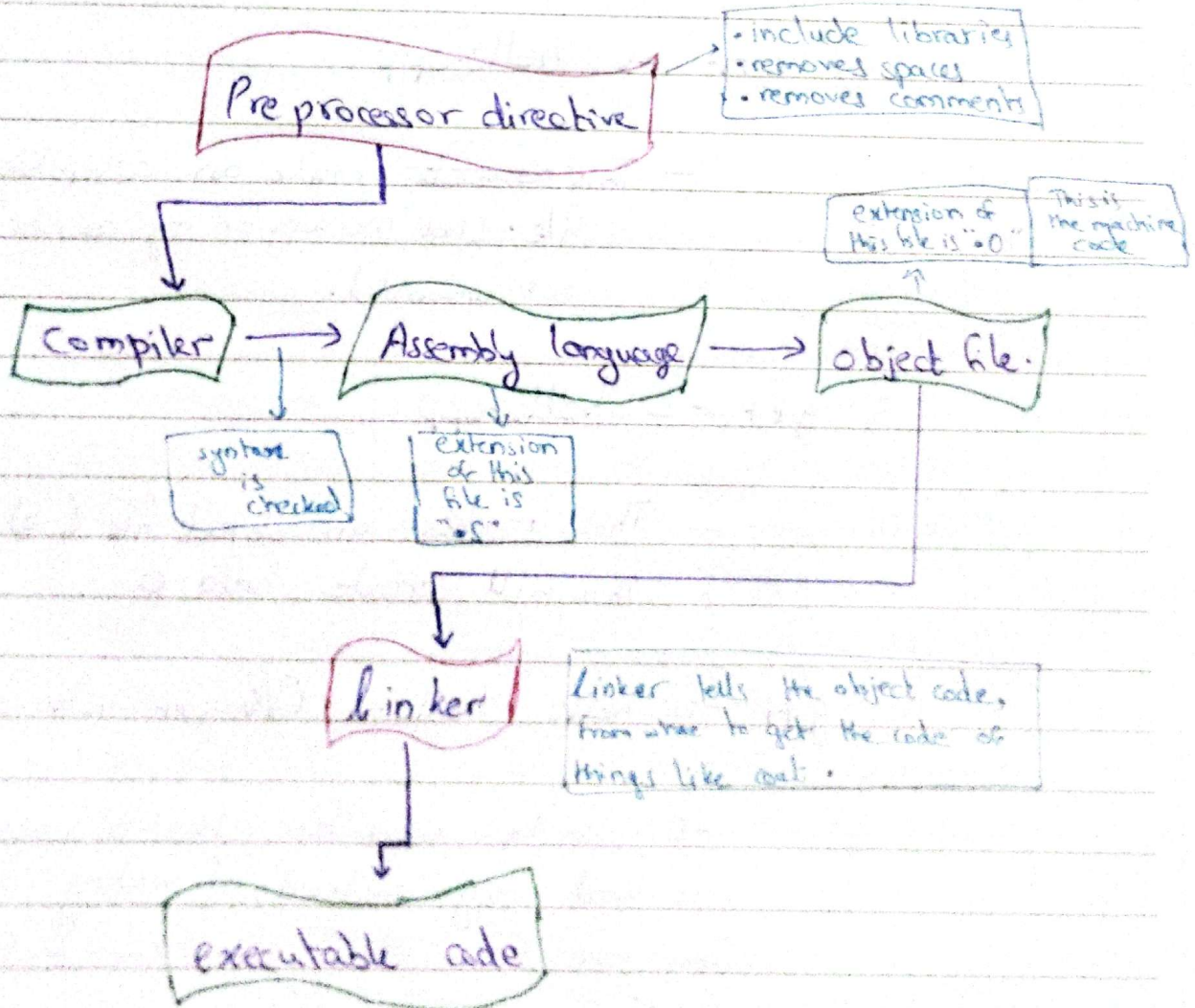
Day

Date

# OOP Object Oriented Programming

Compiling a program in different ways: (ing++).

when we write the command `g++ hello.cpp -o hello`, the following happens.



Yousaf

How to get .s, .o files or show the code of libraries:

1) `g++ -E hello.cpp`

- This shows/displays the whole code including the preprocessor code.

2) `g++ -S hello.cpp`

- This ~~converts~~ make an assemble language file. (not working on my computer as .s is made but unseeable).

3) `g++ -c hello.cpp`

- This makes an object file (.o). This will create hello.o.

4) `g++ -o hello hello.o` OR `g++ hello.o -o hello`

- -o can be used for the 3 above points as well. e.g instead of writing hello (which is an executable file), we can write hello.s or hello.cpp

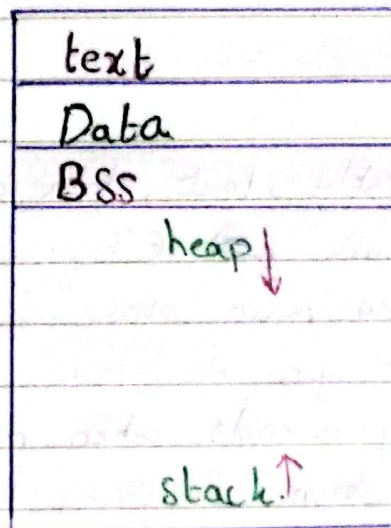
• Remember you can make .s, .o from .cpp but not vice versa. You can make .o from .s but not .s from .o.



Day:

Date: / /

## MEMORY LAYOUT



Ram

Text	Data	BSS (Block started by symbol)
<ul style="list-style-type: none"> <li>- All things are stored here in binary.</li> <li>- This is only read-only.</li> <li>- <del>The</del> <sup>All</sup> const things (variables, arrays etc) are stored in text.</li> <li>- Literals are also stored.</li> </ul>	<ul style="list-style-type: none"> <li>- <u>Initialized</u> global and static variables are stored here.</li> </ul>	<ul style="list-style-type: none"> <li>- <u>Uninitialized</u> global and static variables are stored here.</li> <li>- These <del>are</del> variables are initialized with the <u>null</u> of that type.</li> </ul>

const are added into text with execution of program. e.g. when main is running, ~~const is added~~ <sup>if local const is added</sup> into text.

\* Literals are (abcde... ABCDE) (strings).

- It is the wrong idea that `int main` is the first thing that runs when a program executes.

- When a program<sup>^</sup> executes:

1) Firstly, text, data and BSS are loaded i.e. all ~~global~~ of the global static variable etc.

2) Now main runs.

Main goes into stack.

3) Program ends when main is complete in execution. Everything deletes also text, data and BSS.

\* Where are local variables stored?

- if a local variable is made then it is stored in the stack and if it is initialized then it has garbage values.

- if local variable are const, they are NOT stored in stack. They ARE stored in TEXT.

Some important points:

- Heap is very much larger than stack.

- pointers (will be next topic) even if made for DMA are made in stack and will point to memory in heap (if DMA).

youself



Day:

Date: / /

≡ When Program ends everything deletes with the process last in first out ~~eg~~ i.e. stack is emptied then text, BSS, data are emptied. BUT heap is never deleted so we need to delete it manually (will do in pointers chapter).

- string name = "Hello"; ✓ is a literal.  
char p[] = "Hello"; ✗ not a literal. (so not stored in text)