Create a program

* 1. Variables
* It can be global, local, static, etc
* Global:
  + Unique
  + Defined outside function
  + Have only one copy
  + If it is staic then it is seen only inside the file which it is written in
* Local:
  + It is seen only in the function
  + It is removed when the function leave
  + Static local is permant
  1. Compilation process
* gcc t1.c t2.c -> this would convert c file into executable file
* it is compiler, assembler and then linker

static vs dynamic Linking

* in static the linker would include the needed coded in the .out file
* in dynamic a reference for the code which is needed is made in .out file

Make File

* It is consisted of target, dependencies and rules
* Make uses time stamps of source files to decide which rule to execute
* When u call make file it tries to execute the first target in the file
* U can change the name of the make file by adding -f
  + Make -f mk3 -> would call the make file which name is mk3
* Variables
  + $@ : current target
  + $< : first dependency
  + $ˆ: all dependency
  + $?:list of dependency changed before the current target

Structure:

* It is a collection of data
* Every element should be from known type except self-referencing point as it is always 4 byte

Binary tree

* It has key and left, right pointer where the left is always less than the key and right is greater than the key

process:

Multi tasking

fgh