where the C program stores the data that is to be processed next and the heap is a memory location where the C program stores the data that is to be processed next but the data that is to be processed next is not always stored in the same location as the data that is to be processed next because the C program can keep data in different memory locations depending on the situation and this is why it is important for you to understand the stack and heap because you will be able to understand the C program better and be able to ask more questions on the C program. Alright ... I 'm talking about C program that 's why I say that you get a good picture of memory with the help of C programming therefore , data structures and algorithms is best learned from C and C++ Now here I am dividing it into segments so there is thing called code segment let 's say this is my code let 's say there is a code with the name `` harry.c '' now this code will be first loaded into my main memory I told you what is the first thing that happens ? then it will come to this line it will come to fun1 ( ) , it will execute fun1 ( ) . fun2 ( ) is calling inside fun1 ( ) now the variables that I had created inside it will be created here let 's say I have created `` k '' & `` l '' let me write here : initialize k and l and after that I 'm calling fun2 ( ) so as soon as fun2 ( ) will be called it will say to fun1 ( ) that you wait for a while , I call fun2 ( ) and be back by fetching the value whatever it will return fun1 ( ) replied that you go and get it call fun2 ( ) and get its value .

me where do I give the milk to the milkman because the pointer will store the address of the malloced memory so the milkman will get the milk from the pointer and the pointer will stay there until you return the function or until you delete it. It will say to fun1 ( ) that you 're coming back and now you can start your execution again . If anyone asks you this question, that why heap is used, it can also be done from stack. So that thing is done here with the help of dynamic memory because when does the stack of a function end it ends when the function is returned. I just come back with the value you just wait here. You go and get it and then what will happen, fun function will be called.

this video we will focus on linked list & binary search tree so if you want to know more about these then you should go ahead and watch the video and learn more about these things . If you still think that you want to do all of these things with Python, then I am saying that when you will go for an interview, the person in front of you will expect C, C++, or Java from you. If you don't know C properly, then I have made a 15-hour long video of C with notes.