**Unit 1 :**

Here you will learn about how to take input/output ,how to use conditionals ,Loops and Most important **Functions .**

Submission Deadline : Aug 10th .   
  
(I have given more time as I am assuming this is the first time you are doing Actual coding outside college )   
  
First Steps :   
  
->Install codeblocks , Download either from official site or from this link <https://goo.gl/EcAKOL>  
  
->It’s a very simple Install ,You need to click next->next- > next -> … -> finish .  
  
**How to Run C files in Code blocks :**->Create a new file using File->New file . (Shortcut + Ctrl+shift + N)   
->Save the file using File - >Save or Save As . (Ctrl + S)  
->Type out your code .  
Sample Code :

|  |
| --- |
| #include <stdio.h>  int main(){  pritnf(“Hello this is working”);  return 0; } |

Note : Notice the int main () and return 0 ,statements ,They are mandatory .void main dosent work in codeblocks .And there is also no need of conio.h and getch() inclusion , (**That’s a bad way of stopping program from exiting ) . Code blocks compiler will automatically stop the execution at the end and show you to output window .  
  
Course Content :   
  
->**Simple There will be set of lessons and assignments ,You need to solve and and submit to use (Read how to submit.txt )   
  
->Lessons are where you learn smaller concepts step by step . and assignments are way you actually solve real world problems .  
  
->If you find the content too easy ,Just solve them up and send to me and ask me for new questions ,But don’t discourage your friends who are slow . I strongly feel **hard-work** is more important than IQ , So do at your own pace and don’t worry about others ,Make sure you are submitting before deadlines .   
You need to solve lessons first and then assignments .  
  
Pre –Talk :   
If you feel doing this will not help your career at all ,Feel free to drop out .Every year I try to teach many students but only some (10%-20%) of them actually listen and learn something , I can bet their futures will be 10 times more better than those who dropped out , in 99.9% of cases .   
You might ask why we need to learn every internal working of every function ,Why can we just use which are already built ,This is what most students Lack .Every year we conduct interviews for selection into Mission Rnd ,Out of more than 1000+ students who take part in tests etc ,There would be atleast 90% of them who couldn’t answer Simple questions like How to you reverse a String/LinkedList . Yes there might be some inbuilt functions you can use to achieve the above task ,But what if the interviewers next question is Reverse every K nodes/letters of a linked list . He can twist the question even more , If you depend too heavily on internal functions ,Then you cant be a good product developer . You **atleast** need to know how Basic functions are implemented internally   
  
  
**How to Solve Lessons :   
->Unizip the given folder and open the Files in Code-blocks using File->Open .  
  
->Before Solving each Lesson ,you need to go through the links provided in the C files ,Read them and understand , Don’t skip the Reading part ,If you find Reading very boring , then your basics would be very weak .  
  
->** Read each Lesson file Line by Line ,Read the Questions in them Like what will be the output etc ?   
First Think what can be the answer and Right it down ,in the space provided below main function .   
and then run the C file ,and observe whether you are correct or not . If you are wrong Analyse why you are wrong and write down the correct answer.  
  
->Don’t Just run the files and observe the answers and write them down ,This way you will never learn coding , You never learn before too and don’t repeat the same mistake .  
  
->In some cases you are required to un comment a block of code and run the file again ,Observe whats happening and Answer the questions given .  
  
->You some times need to Solve certain Tasks where you need to write 1-10 lines of code , Feel free to experiment there .   
  
->If you are asked to take input from keyboard ,Make sure you are using scanf(“) or any other input commands and taking the input ,don’t run with default values .  
  
->**LEARN TO COMMENT AND UNCOMMENT PARTS OF CODE .   
  
How to solve assignments :**->Here you will be given a Question and Function to fill ,You need to fill that function and test it with various test cases from main .  
  
->**Functions will be very hard to understand at first ,but trust yourselves , In no time you will be writing high quality code .**   
  
->Don’t try to solve the whole problem at once , Break it into steps by steps and then Solve it . Like for a basic Binary String to Int conversion ,

->Think what would be the sample inputs and outputs ,For “111” its 7 . For “1001” its 9 .   
->How would you solve it with basic common sense on a pen and paper ? Take the last character , Multiply it with 2^0 (2 power 0) .Take last but 1 character multiply it with 2 ^ 1 , And all these intermediate products .  
Ie For “1001” you will do (1\*8)+(0\*4)+(0\*2)+(1\*1) = > 8+0+0+1 => 9   
->Now how do you convert it into a code . First know how to take string as a input . Next Think I need a way to scan the String from right to left , Know how string length is calculated , Next I need to keep a sum variable where I constantly add new products .

|  |
| --- |
| Sample Code would be :  int power\_2=1; for(i=len-1;i>=0;i--){  cur\_char=bin\_str[i];  //You need a way to convert char into int .  cut\_int=cur\_char-‘0’;  sum=sum+(cur\_int \* power\_2);  power\_2=power\_2 \* 2; }  //Final int value will be in sum . |

**And don’t get demotivated if you are not able to code Assignments at first go ,Take mentors help ,Talk to your friends in class who are good and Learn from them ,**There is nothing wrong in asking and learning .   
It’s a very steep hill from Here and your Goals you mentioned in the Form , Do believe in yourself and Just Code . All the best .  
Special Note :   
I observe many students start something ,Do 30% of it , Find it very difficult but wont accept it or be ready to take help , Some how get demotivated and Start one more thing and do again 30% of it , And get demotivated again .   
Every year atleast 10-20 students are like those , Don’t be like that , Everything is hard at some point or other ,There is no point in being a Jack of all trades but master of none . You just need to stay motivated and move past that Hard point , (Either by putting in more work or by learning from friends who already are past that point or taking help from mentors ) , Once you cross that it would be very easy after it . **Even moving from Turbo C to Code-blocks is tough ,but once you are habituated ,you will never touch TurboC again .**  
Last year there are 2-3 students who are not able to write even 2-5 Lines of code at first ,But through sheer hardwork ,They can confidently solve most of the questions in Arrays,Strings ,Linked Lists etc . and maybe write 50-100 Lines of code per day .Just think you too can do it .   
  
**I personally would hire a student who is skilled in 1 area and knows Ins and Outs in that area ,than a student who knows 20% of stuff in 7-8 areas . Its useless .**