**Next Line – CODE SEGMENT**

CODE SEGMENT is the starting point of the Code Segment in a Program and CODE is the name given to this segment and SEGMENT is the keyword for defining Segments, Where we can write the coding of the program.

**Next Line –     ASSUME DS:DATA CS:CODE**

In this Assembly Language Programming, their are Different Registers present for Different Purpose So we have to assume DATA is the name given to Data Segment register and CODE is the name given to Code Segment register (SS,ES are used in the same way as CS,DS )

**Next Line – START:**

START is the label used to show the starting point of the code which is written in the Code Segment. : is used to define a label as in C programming.

**Next Line – MOV AX,DATA  
MOV DS,AX**

After Assuming DATA and CODE Segment, Still it is compulsory to initialize Data Segment to DS register.  MOV is a keyword to move the second element into the first element. But we cannot move DATA Directly to DS due to MOV commands restriction, Hence we move DATA to AX and then from AX to DS. AX is the first and most important register in the ALU unit. This part is also called INITIALIZATION OF DATA SEGMENT and It is important so that the Data elements or variables in the DATA Segment are made accessable. Other Segments are not needed to be initialized, Only assuming is enhalf.

**Next Line – MOV AL,NUM1  
      MOV BL,NUM2**

The above two line code is used to move the two variables i.e. NUM1 and NUM2 to AL and BL registers respectively.

**Next Line – XCHG AL,NUM2  
      XCHG BL,NUM1**

The above two line code is used to Exchange NUM2 variable to AL registers and to Exchange NUM1 variable to BL registers respectively.

Now, we have understood part of it to Exchange to number we can write XCHG NUM1, NUM2, But there is no permutation for **XCHG memory, memory**, Hence we have to send one memory variable to AL or AX depending on DB or DW and send another one memory variable to BL or BX depending on DB or DW. Now we are taking DB, So we have t0 instruction **XCHG** **AL,NUM2** Exchange NUM2 variable value to AL Register in which value of NUM1 variable is already present **XCHG** **BL,NUM1** Exchange NUM1 variable value to BL Register in which value of NUM2 variable is already present. Finally we have Exchanged NUM1 with NUM2 memory variables.

**Next Line – MOV AH,4CH  
INT 21H**

The above two line code is used to exit to dos or exit to operating system. Standard Input and Standard Output related Interupts are found in INT 21H which is also called as DOS interrupt. It works with the value of AH register, If the Value is 4ch, That means Return to Operating System or DOS which is the End of the program.

**Next Line – CODE ENDS**

CODE ENDS is the End point of the Code Segment in a Program. We can write just ENDS But to differentiate the end of which segment it is of which we have to write the same name given to the Code Segment.

**Last Line – END START**

END START is the end of the label used to show the ending point of the code which is written in the Code Segment.