Vidyavardhini's College of Engineering & Technology

Department of Artificial Intelligence and Data Sciene (AI&DS)

Name:	Shagun Upadhyay
Roll No:	60
Class/Sem:	SE/IV
Experiment No.:	8
Title:	Mixed language program to add two numbers
Date of Performance:	22/03/2024
Date of Submission:	05/04/2024
Marks:	
Sign of Faculty:	

Aim: Mixed language program for adding two numbers.

Theory:

C generates an object code that is extremely fast and compact but it is not as fast as the object code generated by a good programmer using assembly language. The time needed to write a program in assembly language is much more than the time taken in higher level languages like C.

However, there are special cases wher a function is coded in assembly language to reduce the execution time.

Eg: The floating point math package must be loaded assembly language as it is used frequently and its execution speed will have great effect on the overall speed of the program that uses it.

There are also situations in which special hardware devices need exact timing and it is must to write a program in assembly language to meet this strict timing requirement. Certain instructions cannot be executed by a C program

Eg: There is no built in bit wise rotate operation in C. To efficiently perform this it is necessary to use assembly language routine.

Inspite of C being very powerful , routines must be written in assembly language to:

1.Increase the speed and efficiency of the routine

2.Perform machine specific function not available in Microsoft C or Turbo C.

3. Use third party routines

Combining C and assembly:

 Built-In-Inline assembles is used to include assembly language routines in C program without any need for a specific assembler.

Such assembly language routines are called in-line assembly.

They are compiled right along with C routines rather than being assembled separately and then linked together using linker modules provided by the C compiler.

Turbo C has inline assembles.

In mixed language program, prefix the keyword asm for a function and write Assembly instruction in the curly braces in a C program
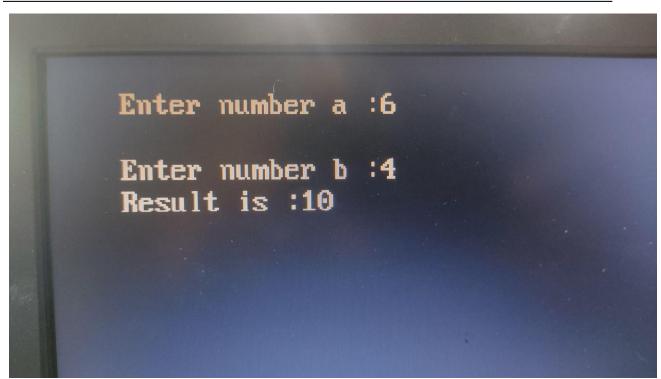
Code :

```c
#include<stdio.h>
#include<conio.h>

void main(){
int a, b, c;
clrscr();
printf("\nEnter number a :");
scanf("%d", &a);
printf("\nEnter number b :");
scanf("%d", &b);
asm{
mov ax, a
mov bx, b
sub ax, bx
mov c, ax
}
printf("Result is :%d", c);

getch();
}
```

Output :

```
Enter number a :6

Enter number b :4
Result is :10
```

Conclusion:

In conclusion, the mixed-language program presented combines assembly language (specifically, 8086 assembly for the EMU8086 environment) with C to add two 16-bit numbers. The program leverages the strengths of both languages: assembly for low-level operations and C for higher-level logic and I/O.

Conclusion:

1. Explain any 2 branch instructions.
2. Explain the syntax of loop.