1) Program to add two numbers in assembly language.

```
. model small
. data
a   db  10
b   db  20
c   db  ?
. code
    mov  ax, @data
    mov  ds, ax
    mov  al, a
    add  al, b
    mov  c, al
    mov  ah, 4ch
    int  21h
    end .
```

2) Program to add two numbers using dw directive.

```
. model small
. data
a   dw  1234
b   db  25
c   dw  ?
. code
        mov  ax, @data
        mov  ds, ax
        mov  bl, b
        add  ax, bx
```

```
mov c, ax
mov ah, 4ch
int 21h
end.
```

3) Program to multiply two numbers for dw directive.

```
.model small
.data
a dw   2345h
b dw      4521h
c dw     ?
d dw     ?
.code
        mov   ax, @data
        mov   ds, ax
        mov   ax, a
        mov   bx, b
        mul   bx
        mov   c, ax
        mov   d, dx
        mov   ah, 4ch
        int   21h
        end
```

4) Program to multiply two numbers (for db directly).

```
.model small
.data
    a  db   10h
    b  db   15h
    c  dw   ?
.code
        mov ax, @data
        mov ds, ax
        mov al, a
        mul b
        mov c, al
        mov ah, 4ch
        int 21h
        end
```

5) Program to solve (a+b) −d, where :−
                a = 2345
                b = 4521
                d = 1256

```
.model small
.data
    a dw   2345h
    b dw   4521h
    d dw   1256h
    c dw   ?
```

```
r dw   ?
.code
      mov  ax, @ data
      mov  ds, ax
      mov  ax, a
      mov  bx, b
      add  bx, ax
      mov  ax, d
      sub  bx, ax
      mov    c, bx
      mov    ah, 4ch
      int    21h
      end
```

6) Program to subtract two numbers (db) (Assembler directive)

```
. model small
. data
a   db  30h
b   db  25h
c   db  ?
. code
      mov  ax, @ data
      mov  ds, ax
      mov  al, a
      mov  bl, b
      sub  al, bl
      mov  c, al
      mov  ah, 4ch
      int  21h
      end .
```

7) Program to solve :-(a+b) * (c+d) / (b+d); a = 245, b = 25, c = 34, d = 50

```
.model small
. data
a   dw   245
b   dw    25
c   dw    34
d   dw    50
r1  dw    ?
r2  dw    ?
r3  dw    ?
quo db    ?
rem db    ?

.code
mov  ax, @ data
mov  dx, ax
mov  ax, a
mov  bx, b
add   ax, bx
mov   r1, ax

mov   ax, c
mov   bx, d
add   ax, bx
mov   r2, ax
```

```
mov ax, b
mov bx, d
add ax, bx
mov r3, ax


mov ax, r1
mul r2
div r3
mov quo, al
mov rem, ah
mov ate ah, 4ch
int 21h
end.
```

8) SWAPPING:-

```
.model small
.data
a db 10h, 20h, 30h, 40h, 50h
b db 11h, 22h, 33h, 44h, 55h
.code
mov ax, @ data
mov ds, ax
lea si, a
lea di, b

mov cl, 05h
L1: mov al, [si]
```

```
xchg  al, [di]
mov   [si], al
inc   si
inc   di
loop  L1
mov ah, 4ch
int 21h
end.
```

9) Perform addition of these two numbers, :- 06034298 &
   7571364.

```
. model small
. data
num 1   db  00h, 03h, 42h, 98h
num 2 db   07h, 57h, 13h, 64h
res   db   4  dup(?)
.code
mov ax, @data
mov  ds, ax
mov si, 03h
mov cl, 04h
clc
deciadd : mov al, num1 [si]
          adc al, num2 [si]
          daa
          mov res [si], al
          dec si
          Loop deciadd
          mov ah, 4ch
          int 21h
          end.
```

10) <u>Use of daa :-</u>

    · model small
    · data
    · code
      mov ax, @ data
      mov ds, ax
      mov al, 63h
      add al, 88h
      daa
      mov ah, 4ch
      int 21h
      end