

```
include irvine16.inc
```

```
.data
```

```
    val byte " apple ",0
```

```
    space byte "   ",0
```

```
.code
```

```
main proc
```

```
    mov ax,@data
```

```
    mov ds,ax
```

```
    ;-----Clear Register-----
```

```
    xor edx,edx
```

```
    xor eax,eax
```

```
    xor ebx,ebx
```

```
    xor ecx,ecx
```

```
    ;-----
```

```
    ;-----get start Program-----
```

```
    ;-----Co-ordinates-----
```

```
        mov dl,0        ;x co -ordinates
```

```
        mov dh,0        ;y co -ordinates
```

```
        push edx
```

```
    ;-----
```

```
        mov bx,2
```

```
        mov eax,100
```

```

;dl=0
;dh,1
.while bx>1 ; ebx>1 (0>1) =True
    mov ecx,23
    forloop:
        call delay
        pop edx
        ;-----
        call gotoxy
        push edx

        ;-----
        ;xor edx,edx
        ;mov edx,offset space
        ;call writestring

        mov ah,40h ; write to file/device
        mov ebx,1 ; output handle
        mov ecx,SIZEOF space ; number of bytes
        mov edx,OFFSET space ; addr of buffer
        int 21h

        ;-----

        pop edx
        add dl,7
        add dh,0

        push edx

```

;-----print space previos location-----

;mov edx,offset val

;call writestring

mov ah,40h ; write to file/device

mov ebx,1 ; output handle

mov ecx,SIZEOF val; number of bytes

mov edx,OFFSET val ; addr of buffer

int 21h

;-----

loop forloop

,,,,,,,,,2nd loop in the while loop-----

mov ecx,20

secloop:

call delay

pop edx

;-----

call gotoxy

push edx

;-----

xor edx,edx

;mov edx,offset val

;call writestring

```
mov ah,40h ; write to file/device
mov ebx,1 ; output handle
mov ecx,SIZEOF val; number of bytes
mov edx,OFFSET val ; addr of buffer
int 21h
```

```
pop edx
sub dl,7
add dh,0
```

```
push edx
```

```
;-----print space previos location-----
```

```
xor edx,edx
;mov edx,offset space
;call writestring
```

```
mov ah,40h ; write to file/device
mov ebx,1 ; output handle
mov ecx,SIZEOF val; number of bytes
mov edx,OFFSET val ; addr of buffer
int 21h
```

```
loop secloop
```

```
.ENDW
```

exit

main endp

end main