

```

;
; Esercizio3, Compito LUG 2013
;
; "Scrivere un programma in Assembly che, data una matrice quadrata M di interi a 32 bit
; stampi su video "Vero" se la diagonale principale di M coincide con la diagonale secondaria
; capovolta e stampi "Falso" altrimenti."
; Segue un esempio:
;
;
;          ebx
;      edx  ->  2      15      3      31
;              4      5      16      11
;              7      5      16      56
;      edi  ->  2      21      4      31
;
; In questo caso il programma stamperà "Vero".

```

```

section .data
    M                dd      2, 15, 3, 31, 4, 5, 16, 11, 7, 5, 16, 56, 2, 21, 4, 31
    dim_M            equ     $-M
    n                equ     4

    MsgTrue          db      "Vero", 10
    LenTrue           equ     $-MsgTrue
    MsgFalse          db      "Falso", 10
    LenFalse          equ     $-MsgFalse

```

```

section .bss
section .text
    global _start

```

```

_start:
    mov     esi, n
    shl     esi, 2                ; esi=dimensione di una riga di M=n*4

    mov     edx, 0                ; edx="indice" prima riga
    mov     edi, dim_M
    sub     edi, esi              ; edi="indice" ultima riga
    mov     ebx, 0                ; ebx=indice colonna

```

```

scan_diags:
    cmp     ebx, n
    jge     print_true

    mov     eax, [edx+ebx*4+M]
    mov     ecx, [edi+ebx*4+M]

    cmp     eax, ecx
    jne     print_false

    add     edx, esi
    sub     edi, esi
    inc     ebx

    jmp     scan_diags

```

```

print_true:
    mov     edx, LenTrue
    mov     ecx, MsgTrue
    mov     ebx, 1
    mov     eax, 4
    int     0x80
    jmp     exit

```

```

print_false:
    mov     edx, LenFalse
    mov     ecx, MsgFalse
    mov     ebx, 1
    mov     eax, 4

```

```
int 0x80
```

```
exit:
```

```
mov     ebx, 0
```

```
mov eax, 1
```

```
int 0x80
```