

30 Gennaio 2020

Elementi di Architetture e Sistemi Operativi

Esercitazione – LC3

- 1) Scrivere un programma assembler LC3 che legge una stringa da tastiera, terminata dal tasto invio (enter) lo salva in una variabile stringa (vettore di caratteri) terminata da '\0' e successivamente conta quanti caratteri è lunga la stringa (ovvero da quanti caratteri è composta '\0' escluso)

```
.ORIG X3000
    ;Viene chiesto di inserire una stringa
    ;Trasforma tutti i caratteri maiuscoli in minuscolo

    LEA    R0, STRINGA    ; PREDISPONGO LA DOMANDA
    PUTS

    LEA    R3, STRING
    LD     R2, LF    ; Load 10
```

AGAIN:

```
    GETC                ; Request keyboard
    OUT                 ; Stampo il carattere a video
    ADD    R1,R2,R0     ; Test for terminating
    BRz    EXIT        ; character
    STR    R0, R3, #0
    ADD    R3, R3, #1
```

```
    BRnzp AGAIN    ; ... Continuo all'infinito
```

```
EXIT: LD R0, ZERO
      STR R0, R3, #0    ; Termino la stringa letta
```

CONTA:

```
    LEA    R0, STRING
    LD     R2, ZERO
    LD     R4, ZERO    ; CONTA I CARATTERI
```

INICONTA:

```
    LDR    R1, R0, #0
    ADD    R3, R1, R2    ; Ho trovato \0 sulla stringa
    BRz    FINECONTA
    ADD    R0, R0, 1
    ADD    R4, R4, 1
    BRnzp INICONTA
```

FINECONTA:

```
    LEA    R0, RISPOSTA    ; PREDISPONGO LA RISPOSTA
    PUTS

    LD     R1, ZEROCHAR
```

```
ADD R0, R4, R1
OUT
HALT
```

```
; Definizione di variabili
```

```
STRINGA .STRINGZ "Digitare una parola: " ; Stringa già terminata con \0
```

```
RISPOSTA .STRINGZ "La stringa è lunga: "
```

```
STRING .BLKW 10 x0042
```

```
LF .FILL xFFF6
```

```
ZERO .FILL x0000
```

```
ZEROCHAR .FILL 48
```

```
.END
```

- 2) Si modifichi il programma precedente, trasformando in funzione il conteggio della lunghezza della stringa. Il main si dovrà occupare di leggere da tastiera la stringa, invocare la funzione che determina la lunghezza della stringa e successivamente di stampare a video la lunghezza della stringa.

- 3) Scrivere una funzione LC3 che preso in input in R1 un valore decimale (non espresso in ascii ma proprio decimale/binario) < 10 (0..9) stampi a video il conto alla rovescia da quel valore a 0.

Es: in R1=4

STAMPA: 43210

Si scriva inoltre un programma assembler LC3 che legge un numero < 10 (0 .. 9) da tastiera, e richiami la funzione scritta in precedenza per stampare a video il conto alla rovescia.

.ORIG X3000

AGAIN:

```
AND    R1,R1,0      ; Azzero R1
LD     R2,ZERONEG   ; R2 <= ASCII NEG

TRAP   x23          ; Request ke yboard

ADD    R1,R0,0      ; R1 <= R0
ADD    R1,R1,R2     ; trasformo il valore ASCII in binario
                          ; ADESSO R1 contiene un valore binari/decimale NON ASCII

JSR    COUNTDOWN

TRAP   x25
```

COUNTDOWN:

```
ADD    R6,R7,0      ; Mi salvo il valore di ritorno TRAP Sporca il valore per tornare
LD     R3,ZEROPOS   ; R3 <= ASCII POS
```

```
INIZIO: ADD    R0,R1,R3 ; R1 conntiene il valore in decimale, R3 Contiene il DELTA per ASCII
                          ; Con la SOMMA metto in R0 il valore ASCII da stampare
TRAP   X21            ; Stampo a video il valore
```

```
ADD    R1,R1,-1     ; Decremento R1
BRZP   INIZIO       ; fino a che non diventa negativo R1
```

```
ADD    R7,R6,0      ; Ripristino il return Value
RET
```

```
ZERONEG .FILL -48 ; Codice Ascii 0 da sottrarre
ZEROPOS .FILL 48  ; Codice ASCII 0 da sommare
UNO     .FILL -1  ; Decremena
```

.END

- 4) Si scriva la funzione LC3 che presa in input in R0 l'indirizzo di una stringa, ne faccia il lowercase (ovvero dei SOLI caratteri in maiuscolo, vengano trasformati in minuscolo) la stringa in R0 verrà modificata (non una sua copia)
 Scrivere inoltre un programma assembler LC3 che legge una stringa da tastiera stampandola a video, terminata dal tasto invio (enter) la salva in una variabile stringa (terminata da '\0') richiama la funzione tolower (scritta in precedenza) e ristampa la stringa a video.

```
.ORIG X3000
    ;Viene chiesto di inserire una stringa
    ;Trasforma tutti i caratteri maiuscoli in minuscolo

    LEA    R0, STRINGA    ; PREDISONGO LA DOMANDA
    PUTS

    LEA    R3, STRING
    LD     R2, LF        ; Load 10
```

AGAIN:

```
    GETC                ; Request keyboard
    OUT                 ; Stampo il carattere a video
    ADD    R1,R2,R0     ; Test for terminating
    BRz    EXIT        ; character
    STR    R0, R3, #0
    ADD    R3, R3, #1

    BRnzp AGAIN    ; ... Continuo all'infinito
```

```
EXIT: LD R0, ZERO
    STR R0, R3, #0    ; Termino la stringa letta

    LEA    R0, STRING    ; Metto indirizzo in R1

    JSR    TOWER

    LEA    R0,STRING
    PUTS

    HALT
```

```
TOWER: ;R0 <- Indirizzo Stringa
    LD R3, ANEG
    LD R4, ZNEG
    LD R5, UPLOW
```

inizioW:

```
LDR R1, R0, #0    ; R1 <- STRING[i]
BRnz fineW      ; E' finita la stringa
```

```
ADD R2, R1, R3
BRn increment    ; Se il carattere viene prima della A
```

```
ADD R2, R1, R4
BRp increment    ; Se il carattere viene DOPO della Z
```

```
ADD R1, R1, R5
STR R1, R0, #0    ; Salvo il carattere modificato
```

increment:

```
ADD R0, R0, #1    ; INCREMENTO IL PUNTATORE
BRnzp inizioW
```

fineW: RET

; Definizione di variabili

STRINGA .STRINGZ "Digita una frase (Invio per terminare): " ; Stringa già terminata con \0

STRING .BLKW 20

;CARATTERE .STRINGZ "Digita il carattere da cercare: "

LF .FILL xFFF6

ASCII .FILL 48

ZERO .FILL 0

UPLOW .FILL 32

ANEG .FILL xFFBF ; -65 -A

ZNEG .FILL xFFA6 ; -90 -Z

.END