

Esercizi relativi Macchina standard MAC-1

1. Si deve progettare una CPU con un parallelismo di 16 bit. Si possono scegliere alcune alternative per il formato delle istruzioni, quali ad esempio:

Codice Operativo (CO) (bit)	Indirizzo Operando < IND > (bit)
3	13
4	12
5	11

Discutere vantaggi e svantaggi delle varie alternative.

2. Scrivere uno spezzone di programma in linguaggio MAC-1 che somma i numeri interi da 1 a 32, e mette il risultato nella locazione di memoria 1064.

Soluzione: Con ciclo a condizione finale:

```

max=32                / costante
sum : 1064            / indirizzo somma
cont: 1065           / indirizzo contatore

      LOCO  0
      STOD  sum          / azzera somma
      STOD  cont        / azzera contatore

INI:   LOCO  1          / ac <-- 1
      ADDD  cont        / ac <-- cont+1
      STOD  cont        / memorizza contatore
      ADDD  sum         / ac <-- sum+cont
      STOD  sum         / memorizza somma
      LODD  cont
      SUBD  max         / ac <-- cont-max : controllo ciclo
      JNEG  INI        / se negativo va a INI
      HALT
  
```

Soluzione: Con ciclo a condizione iniziale:

```

max=32                / costante
sum : 1064            / indirizzo somma
cont: 1065           / indirizzo contatore

      LOCO  0
      STOD  sum          / azzera somma
  
```

```

        STOD  cont                / azzera contatore
INI:   LOCO  1                    / ac <-- 1
        ADDD  cont                / ac <-- cont+1
        STOD  cont                / memorizza contatore
        SUBD  max                  / ac <-- cont-max : controllo ciclo
        JPOS  FIN                  / se >=0 va a FIN
        ADDD  sum                  / ac <-- sum+cont
        STOD  sum                  / memorizza somma
        JUMP  INI
FIN:   HALT

```

3. Descrivere in linguaggio MAC-1 un algoritmo che dati 2 interi nelle locazioni $A = 1024$ e $B = 1023$, pone sullo stack il $\min[A, B]$.

Soluzione:

```

a : 1024                / indirizzo di a
b : 1023                / indirizzo di b

        LODD  a                    / ac <-- a
        SUBD  b                    / ac <-- a-b
        JNEG  AMIN                 /
        LODD  b                    / ac <-- b
        PUSH                     / mette b sullo stack
        JUMP  FIN
        LODD  a                    / ac <-- a
AMIN:  PUSH                     / mette a sullo stack
FIN:   .....

```

4. Scrivere una procedura in linguaggio MAC-1 che riceva in input un indirizzo di memoria e restituisca un intero che rappresenti il numero di bit uguali a 1 nella stringa binaria contenuta nella locazione di memoria puntata dal parametro in input.

Soluzione: Un test sul valore del bit più significativo di una stringa binaria interpretata come numero intero in complemento a 2 può essere realizzato come test sul segno. E' possibile far scorrere ad uno ad uno i bit di una stringa binaria nella posizione più significativa tramite uno shift a sinistra (realizzato come somma del numero con se stesso).

```

c=0                / ind loc contatore di 1
m=1                / ind loc valore variabile
a=3                / ind loc indirizzo variabile

(B1:)  DESP  2                /prera spazio per var loc
        LOCO  0
        STOL  c                / azzera contatore
        LODL  a                / ac <-- ind. di a
        PSHI                     / push valore di a
        POP                      / pop di a
L1:    STOL  m                / ac <-- m (mette valore di a in m)
        JZER  FIN              / se 0 va a FIN

```

```
JPOS  L2          / se >=0 va a L2
                          / m<0 cifra piu, signif =1

LOCO  1
ADDL  c          / ac <-- c+1
STOL  c          / memorizza c
LODL  m          / ac <-- m
L2:   ADDL m      / ac<-- m+m : shift di m
      JUMP L1
FIN:  LODL c      / ac<-- c prepara il return
      INSP 2
      RETN
```