

```

1 .GLOBAL _main
2 .INCLUDE "C:/amb_GAS/utility"
3
4 .TEXT
5 _main:                NOP
6
7                        XOR %ECX, %ECX
8
9                        # Primo numero
10                       CALL lettura_numero # Richiedo il numero
11                       CALL verifica_neg # Incremento CL se necessario
12                       MOV %AH, %BH
13                       MOV %AL, %BL
14
15                       # Vado a capo
16                       CALL newline
17
18                       # Secondo numero
19                       CALL lettura_numero # Richiedo il numero
20                       CALL verifica_neg # Incremento CL se necessario
21
22                       # Vado a capo
23                       CALL newline
24
25                       # A questo punto possiamo fare, in modo rudimentale, BH_BL +
AH_AL
26                       ADD %BL, %AL
27                       CMP $9, %AL
28                       JBE no_carry1
29
30                       INC %CH # Setto il riporto uscente ad 1
31                       SUB $10, %AL # Risultato della somma decrementato in caso di
riporto uscente settato
32
33 no_carry1:            ADD %BH, %AH
34                       ADD %CH, %AH
35
36                       XOR %CH, %CH # Resetto il riporto uscente dopo averlo usato
37
38                       CMP $9, %AH
39                       JBE no_carry2
40
41                       # Non mi serve il riporto uscente della seconda colonna, quindi
non faccio incrementi in nessun registro
42                       SUB $10, %AH # Risultato della somma decrementato in caso di
riporto uscente settato
43
44 no_carry2:            # A questo punto dobbiamo verificare che il risultato sia di
segno positivo o negativo, e tirare le conclusioni
45                       # La somma si trova in AH_AL
46                       # Riprendiamo il registro %CL e verifichiamo quanto trovato
47
48                       # Verifico se i numeri sommati sono discordi, in quel caso non
ho problemi e stampo
49                       CMP $1, %CL
50                       JE stampa_risultato
51
52                       # Verifico, se ho due numeri positivi sommati, che il risultato
sia positivo. Se non lo e' termino l'esecuzione del programma
53                       CMP $0, %CL

```

```

54         JNE verifica_concneg
55         CMP $5, %AH
56         JB stampa_risultato
57         JMP fine
58
59 verifica_concneg:  # Non faccio confronti, sono certo che il valore di CL e' due a
                    # questo punto
60                   # Verifico, se ho due numeri negativi sommati, che il risultato
                    # sia negativi. Se non lo e' termino l'esecuzione del programma
61                   CMP $5, %AH
62                   JB fine
63                   JMP stampa_risultato
64
65 stampa_risultato:  NOP
66
67                   XCHG %AH, %AL
68                   CALL outdecimal_byte
69
70                   XCHG %AH, %AL
71                   CALL outdecimal_byte
72
73                   CALL newline
74                   CALL newline
75                   JMP _main
76
77 fine:             XOR %EAX, %EAX
78                   RET
79
80 /* Sottoprogramma per la lettura di un numero in base 10 a due cifre
81 con adeguati controlli. Pongo il numero letto in AH_AL */
82 lettura_numero:   CALL inchar
83                   CMP '$0', %AL
84                   JB lettura_numero
85                   CMP '$9', %AL
86                   JA lettura_numero
87                   AND $0x0F, %AL
88                   CALL outdecimal_byte
89                   MOV %AL, %AH
90
91 lettura_numero_2: CALL inchar
92                   CMP '$0', %AL
93                   JB lettura_numero_2
94                   CMP '$9', %AL
95                   JA lettura_numero_2
96                   AND $0x0F, %AL
97                   CALL outdecimal_byte
98
99 fine_programma:   RET
100
101 /* Sottoprogramma che incrementa CL se il numero inserito e' negativo.
102 Guardo il registro AH, ricordandomi che a_{n-1} < \beta/2 se voglio avere un numero
    positivo */
103 # Conclusione: CL = 0 numeri positivi, CL = 1 numeri discordi , CL = 2 numeri
    negativi
104 verifica_neg:     NOP
105                   CMP $5, %AH
106                   JB fine_verifica
107                   INC %CL
108 fine_verifica:     RET
109

```