

```
# Scrivere un programma che si comporta come segue:
# 1) stampa un punto interrogativo e legge con eco da tastiera
#    un numero naturale A in base 10, sotto l'ipotesi che stia su 16 bit
# 2) se il numero è minore o uguale ad 1, termina
# 3) altrimenti, stampa la sua scomposizione in fattori primi,
#    cioe' una lista di righe x^y, intendendo che il fattore primo x
#    e' contenuto in A alla sua potenza y-esima, con x>=2, y>=1.
# 4) attende la pressione di un tasto e termina.
#
# Esempio:
#
#?338
#2^1
#13^2
#
#?199
#199^1
#
#?1
#[terminazione]
```

```
_main:      NOP

inizio:     MOV $'?',%AL
            CALL outchar
            CALL indecma1_short
            CALL newline

            CMP  $1,%AX
            JBE  termina

# AX contiene il numero da dividere (inizialmente A) su 16 bit. BX contiene il divisore da
# inizializzare a 2. CX contiene l'esponente del divisore. SI contiene A

ciclo:      MOV  $2,%BX
            CMP %BX, %AX
            JB  termina
            MOV $0,%CX

ciclo2:     MOV  $0,%DX          # [DX,AX] contiene il numero A su 32 bit
            MOV  %AX,%SI        # salvo AX in SI perche' la divisione lo sporca.
            DIV  %BX
            CMP  $0,%DX          # DX contiene il resto della divisione
            JNE  prossimo        # A ha BX come divisore
            INC  %CX
            JMP  ciclo2

prossimo:   MOV %SI, %AX
            CMP $0, %CX
            JE  fine

            PUSH %AX            #stampa un fattore primo su una nuova riga
            MOV %BX, %AX
            CALL outdecimal_short
            MOV $'^', %AL
            CALL outchar
            MOV %CX, %AX
            CALL outdecimal_short
            CALL newline
            POP  %AX

fine:       INC  %BX
            JMP  ciclo

termina:    RET

.INCLUDE "C:/GAS/utility"
```