

Status progetto ASM 22/05/2021

- Prototipo 6, 7 e 8
- itoa (converte numero in stringa)

Fatto

- Scritta la funzione `itoa()` in assembly che converte un numero in stringa e i suoi unit test.

bisogna passare puntatore all'array output e il valore da convertire

- Modificato l'ordine dei parametri della funzione assembly `divisione()`
- Scritti nuovi prototipi in python:
 - il prototipo 6 non necessita piu' di scorrere avanti e indietro la stringa: la stringa viene letta andando semplicemente in avanti carattere per carattere
 - il prototipo 7 suddivide la funzione `postfix()` in una funzione `write_result()` che scrive il risultato dell'espressione in input se sta all'interno dei 10 caratteri
 - il prototipo 8 usa un approccio piu' semplice per riconoscere se il risultato della espressione e' rappresentabile in 10 caratteri (funzione `write_result()`)

Pseudo-codice del concetto:

```
if risultato > limite_superiore or risultato < limite_inferiore:  
    invalid = True
```

In piu' questa funzione permette di evitare di dover ripetere il codice nella funzione `postfix()` necessario per scrivere "Invalid" nel file in output.

Il prototipo 8 aggiunge anche funzioni scritte precedentemente in assembly per cercare di rendere il piu' identici possibili prototipo e progetto finito

Da fare

- [OBIETTIVO PRINCIPALE] completare `postfix.s` (idealmente con l'algoritmo del **prototipo 8**)
- scrivere la funzione `write_result()` in assembly.
- scrivere test per testare l'intero programma (end to end tests)
- aggiungere altri unit test dove sara' necessario
- cercare di ottimizzare l'algoritmo di postfix dove possibile
- scrivere la documentazione

Note aggiuntive

- Le variabili di assembly nei file con le funzioni non vengono settate al valore iniziale quando le funzioni vengono richiamate dalla seconda volta in poi