

Ho pensato all'algoritmo che dovrebbe permetterci di scrivere il codice:

NOTA: dobbiamo decidere se scrivere prima codice C (per avere una linea guida e sapere che l'algoritmo funziona) o direttamente assembly.

- scorrere la stringa in input
- riconoscere dalla stringa gli elementi perché sono separati tra di loro da spazi
- valutare se l'elemento è formato da caratteri validi (`-`, `+`, `*`, `/`, `0-9`) oppure no (lettere o caratteri speciali non riconosciuti)
  - se contiene almeno un carattere non riconosciuto/valido: restituire la stringa "Invalid" al main (C)
  - se i caratteri sono validi distinguere se è operando o operatore

Proseguo qua sotto descrivendo i casi con caratteri validi...

- se è operando convertire i caratteri in numero e metterlo nello stack (push)

ATTENZIONE: bisogna gestire `-` che indica numero negativo

OSSERVAZIONE: il prof ha scritto che nei suoi test "Solo gli operandi negativi hanno il segno riportato esplicitamente in testa" quindi se troviamo il simbolo `+` davanti ad un numero possiamo scegliere se considerarlo valido oppure no... l'importante è che non ci affidiamo al fatto che sarà sempre presente il segno per distinguere se il numero è positivo o negativo (perché appunto non sarà presente nei numeri positivi testati dal prof)

- se è operatore bisogna estrarre (pop) dallo stack i due operandi, fare il calcolo e poi mettere (push) nello stack il risultato
  - il primo operando estratto è l'operando destro
  - il secondo operando estratto è l'operando sinistro
- Convertire l'ultimo elemento dello stack (risultato dell'espressione) in stringa
- Restituire il risultato sotto forma di stringa al main (C)

Alcune di queste informazioni le ho trovate sul convertitore che ci ha dato il prof qui: <https://www.free-online-calculator-use.com/postfix-evaluator.html#how-eval-postfix>