**Pseudocodice di postfix**

**ELABORAZIONE INPUT**

carattere=primo carattere del file //poi nel loop si incrementerà l’indice

If((carattere!= \0 *or* carattere!=\n) *and* invalid==false): proseguo

else: salto a SCRITTURA OUTPUT//l’input è terminato oppure c’è un carattere invalido

push carattere //salvo il carattere per farlo analizzare dalla funzione is\_valid\_char

chiamo is\_valid\_char //funzione spiegata dai diagrammi di flusso

if(invalido): invalid==true e salto a postfix\_incrementa\_indice

else: proseguo

//ora devo capire che tipo di carattere è

if(carattere è operatore): salto a postfix\_is\_operator //chiamata della funzione is\_operator

If(carattere è cifra): push carattere convertito // chiamata della funzione is\_operand, poi verranno controllate anche le cifre successive e moltiplicate via via per 10

if(carattere è segno di numero negativo): is\_negative=true

if (carattere è spazio): push numero salto a postfix\_incrementa\_indice// controllando is\_negative caspisco se rendere il numero positivo o negativo

**postfix\_is\_operator**

if(non ci sono sufficienti elementi nella pila): salto a postfix\_not\_enough\_operands

else: chiamo la funzione dell’operazione corrispondente e poi salto a postfix\_incrementa\_indice

**postfix\_not\_enough\_operands**

invalid=true //dato che non ci sono abbastanza numeri nello stack, la formula non è ben formata

salto a postfix\_incrementa\_indice

**postfix\_incrementa\_indice**

incremento l’indice e salto all’inizio

**SCRITTURA OUTPUT**

//il risultato è nella pila

If(invalid==true): salto a postfix\_write\_invalid\_invchar //chiamerà write\_result che scriverà Invalid

else: salto a postfix\_write\_appearently\_valid\_result //non so ancora se è tutto corretto

**postfix\_write\_appearently\_valid\_result** // controlla se l’espressione è ben formata

if(troppi elementi nella pila): salto postfix\_rm\_too\_many\_els\_stack

else: chiamo write\_result //la formula è ben formata, quindi scrivo il risultato e concludo

**postfix\_rm\_too\_many\_els\_stack**

loop(se ci sono troppi elementi nello stack): rimuovo uno a uno gli elementi e dopo proseguo

**postfix\_write\_invalid\_too\_many\_els\_stack**

scrivo “Invalid” perché l’espressione non era ben formata

**NOTA BENE**

Questo pseudo codice non è una copia esatta e fedele del reale programma dato che sono stati omessi alcuni controlli (es: numeri a più cifre) al fine di rendere più comprensibile la relazione. Le funzioni già spiegate dai diagrammi non sono state approfondite in questa sezione. L’etichetta “fine”, che si occupa di ripristinare i vari elementi dalla pila e eseguire la return, non è stata inserita poiché si assume che si salti a quell’etichetta ogni qualvolta si termini di scrivere il risultato finale.