

# Η Χρήση της Δομής Στοιβάς ( The use of Stack )

(Εργαστήριο 4)

## Περίληψη

- Ορισμός της Στοιβάς
- Η λειτουργία της Στοιβάς (γενικά, αλλά και στο SPIM)
- Η χρησιμότητα της Στοιβάς στους Μικροεπεξεργαστές
- Παράδειγμα
- Παρουσίαση Άσκησης 3 (Δημιουργία και Χρήση της Στοιβάς)

## Δομή Στοιβάς (stack)

Γενικά, ως στοιβα ορίζεται μια δομή δεδομένων η οποία ακολουθεί την πολιτική Last-In-First-Out (LIFO).

Με άλλα λόγια, τα δεδομένα τοποθετούνται σε αυτή με συγκεκριμένη σειρά και αφαιρούνται ή τυγχάνουν διαχείρισης με την ακριβώς αντίστροφη σειρά. Κάθε δομή stack χαρακτηρίζεται πλήρως από δύο δείκτες και τις βασικές λειτουργίες που εκτελεί.

Οι δύο βασικοί δείκτες σημειώνουν τα όρια της και είναι:

- Ο δείκτης βάσης (base pointer – bp) ο οποίος δείχνει το αρχικό στοιχείο της δομής, δηλαδή το πρώτο στοιχείο που μπήκε στην στοιβα και το τελευταίο που θα βγει, και
- Ο δείκτης στοιβάς (stack pointer – sp) ο οποίος δείχνει το επόμενο στοιχείο που θα τύχει διαχείρισης, αυτό δηλαδή που μπήκε τελευταίο στη δομή.

Κατά την αρχικοποίηση της στοιβάς ισχύει ότι  $bp = sp$ .

## Οι Βασικές Λειτουργίες μιας Στοιβάς

- Οι βασικές λειτουργίες διαχείρισης μιας στοιβάς είναι οι εντολές **push** (σπρώχνω) και **pop** (αφαιρώ).
- Η εντολή push είναι η εντολή που «γεμίζει» την δομή με δεδομένα, ακολουθώντας την λογική LIFO.
- Κάθε εντολή push(a) μειώνει τον sp κατά ένα (ή κατά 4) και τοποθετεί στην θέση αυτή το δεδομένο που περιγράφεται (ή δείχνεται, αναλόγως της εφαρμογής) από το a.
- Η εντολή pop είναι η εντολή που «αδειάζει» την δομή από τα δεδομένα. Κάθε εντολή pop(b) αφαιρεί από την στοιβα το στοιχείο που δείχνει ο sp, το τοποθετεί στο b (ή στην θέση που δείχνει ο b, αναλόγως εφαρμογής) και αυξάνει τον sp κατά ένα (ή κατά 4).

## Η Στοίβα στο SPIM

Στην περίπτωση του SPIM, αλλά και όλων των επεξεργαστών, υπάρχει ένας χώρος στην μνήμη (διαδοχικές θέσεις μνήμης) ο οποίος καλείται stack και γενικά διαθέτει το βασικό χαρακτηριστικό της στοίβας (είναι δηλαδή LIFO). Πέραν τούτου, η stack αυτή έχει ειδική αποστολή που αφορά τον χρόνο εκτέλεσης του προγράμματος και ΔΕΝ περιλαμβάνει τις βασικές λειτουργίες που περιγράφονται στην προηγούμενη παράγραφο.

Ο βασικός λόγος ύπαρξης της stack είναι για την αποθήκευση των περιεχομένων των προσωρινών καταχωρητών κατά την κλήση μιας συνάρτησης στον κώδικα. Επιπλέον, είναι ο χώρος όπου αποθηκεύονται τα ορίσματα που δίνονται από την γραμμή εντολών του προγράμματος.



Πανεπιστήμιο Κύπρου

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

## Η Στοίβα στο SPIM

Η Στοίβα στο SPIM ξεκινά (bp) από την τελευταία θέση μνήμης (0x7FFFFFFC) και εκτείνεται προς τα κάτω, κατά σύμβαση. Με άλλα λόγια, ο sp δείχνει πάντα σε μια θέση μνήμης και κάθε φορά που προστίθεται ένα στοιχείο στη stack αυτός μειώνεται κατά 1 θέση (4 αν έχουμε να κάνουμε με words).

Ακριβώς επειδή η stack στο SPIM είναι χώρος μνήμης, για την υλοποίηση δομής στοίβας πρέπει να παρθούν κάποιες σχεδιαστικές αποφάσεις. Αρχικά, πρέπει να ορίσουμε ένα χώρο στην μνήμη του SPIM ο οποίος να μπορεί να χρησιμοποιηθεί έτσι ώστε να φτιάξουμε την δομή.

Αυτός ο χώρος μπορεί να είναι η ίδια η Στοίβα του SPIM, αλλά αυτό δεν είναι απαραίτητο, ενώ σε κάποιες περιπτώσεις μπορεί να αποδειχθεί και επικίνδυνο (όταν π.χ. εμπλακεί με κώδικα κλήσης συναρτήσεων). Από την άλλη, μπορεί να αποδειχθεί χρήσιμη η ύπαρξη του καταχωρητή \$sp καθώς και η σύμβαση της προς τα κάτω αύξησης της μνήμης.



Πανεπιστήμιο Κύπρου

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

## Παράδειγμα Λειτουργίας μιας Στοίβας

1. push(a) # push register a on the stack
2. push(b) # push register b on the stack
3. pop(a) # pop value from the stack to reg. a
4. pop(b) # pop value from the stack to reg. b

\$bp = 0x20000000

\$sp = 0x20000000

Ποιο θα είναι το αποτέλεσμα των πιο πάνω εντολών (σε σχέση με το περιεχόμενο των καταχωρητών);



Πανεπιστήμιο Κύπρου

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

## Άσκηση 3

Σε αυτή την άσκηση καλείστε να υλοποιήσετε έναν αλγόριθμο, ο οποίος θα δέχεται από την γραμμή εντολών μία εξίσωση (χωρίς κενά). Στη συνέχεια, με την υλοποίηση μιας δικής σας Στοίβας (σε περιοχή της μνήμης που είναι εκτός της καθορισμένης από το SPIM περιοχής) και των δικών σας συναρτήσεων push και pop θα πρέπει να κάνετε χρήση της Στοίβας και να ελέγχετε εάν τα ζεύγη παρενθέσεων όλων των τύπων { [ ( ) ] } βρίσκονται σε σωστή αλληλουχία. Για την υλοποίηση της άσκησης που σας ανατίθεται πρέπει να καθορίσετε άλλο χώρο στη μνήμη πέραν της καθορισμένης Στοίβας του SPIM. Κατά συνέπεια, πρέπει να ορίσετε:

1. Τα όρια της μνήμης που θα κατέχει η δομή,
2. Κάποιο καταχωρητή σαν \$bp και κάποιον σαν \$sp
3. Το μέγιστο μέγεθος της Στοίβας
4. Επίσης, θα πρέπει να υλοποιηθούν οι βασικές λειτουργίες της δομής (push, pop) καθώς και οι ζητούμενες πράξεις.



Πανεπιστήμιο Κύπρου

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

## Ελάχιστες Προδιαγραφές Κώδικα

1. Το πρόγραμμα πρέπει να δέχεται μια εξίσωση (σε μορφή string) από την γραμμή εντολών, η οποία θα αποθηκεύεται στη μνήμη (όπως και στην άσκηση 2).
2. Ο κώδικας σας θα ελέγχει ένα-ένα τα ψηφία που δόθηκαν από τη γραμμή εντολών και **α)** όταν ανιχνεύει οποιαδήποτε αριστερή παρένθεση (δηλαδή άνοιγμα παρένθεσης) να πραγματοποιεί push (εάν η στοίβα βρίσκεται εντός των προκαθορισμένων ορίων), **β)** όταν ανιχνεύει δεξιά παρένθεση να κάνει pop για να ελέγξει εάν κλείνει κανονικά κάποιο ζεύγος παρενθέσεων.
3. Οι συναρτήσεις push και pop πρέπει να διαχειρίζονται λέξεις των 32 bits.
4. Η Στοίβα θα μπορεί να κρατά μέχρι 8 παρενθέσεις ανά πάσα στιγμή. Αν χρειαστεί να αποθηκευτεί 9η παρένθεση, τότε θα εκτυπώνεται μήνυμα ότι η Στοίβα είναι εκτός ορίων και το πρόγραμμα θα τερματίζει.
5. Στο τέλος του προγράμματος θα πρέπει να εκτυπώνεται ο συνολικός αριθμός των σωστών παρενθέσεων μέχρι το σημείο που εντοπίστηκε το λάθος (νοούμενου ότι υπάρχει κάποιο λάθος).
6. Πρέπει να χρησιμοποιηθούν μόνο οι καταχωρητές \$a0, \$a1 και \$v0 για μεταφορά τιμών προς και από τις συναρτήσεις push και pop.



Πανεπιστήμιο Κύπρου

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

## Διάγραμμα Ροής Άσκησης 3

Φτιάξτε το δικό σας διάγραμμα ροής για αυτή την άσκηση!  
(5 λεπτά)



Πανεπιστήμιο Κύπρου

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

## Μαθησιακοί Στόχοι – Εργαστήριο 4

1. Ορισμός και λειτουργία της Στοίβας
2. Ποια η χρησιμότητα της Στοίβας στους Μικροεπεξεργαστές
3. Δημιουργία και Χρήση της Στοίβας για επίλυση προβλημάτων.



Πανεπιστήμιο Κύπρου

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών