

Εργασία 4 – Περιβάλλον καταναεμημένης εκτέλεσης με διαφανή μετανάστευση και εξισορρόπηση φορτίου**Περιβάλλον εκτέλεσης SimpleScript**

Υλοποιήστε ένα περιβάλλον εκτέλεσης για προγράμματα στην γλώσσα SimpleScript, με την παρακάτω σύνταξη:

Program = Tag {InstrLine}.		
InstrLine = [Label] Instr.		
Instr = SET Var VarVal		Var = VarVal
ADD Var VarValI1 VarValI2		Var = VarValI1 + VarValI2
SUB Var VarValI1 VarValI2		Var = VarValI1 - VarValI2
MUL Var VarValI1 VarValI2		Var = VarValI1 * VarValI2
DIV Var VarValI1 VarValI2		Var = VarValI1 / VarValI2
MOD Var VarValI1 VarValI2		Var = VarValI1 % VarValI2
BGT VarValI1 VarValI2 Label		if (VarValI1 > VarValI2) goto Label
BGE VarValI1 VarValI2 Label		if (VarValI1 >= VarValI2) goto Label
BLT VarValI1 VarValI2 Label		if (VarValI1 < VarValI2) goto Label
BLE VarValI1 VarValI2 Label		if (VarValI1 <= VarValI2) goto Label
BEQ VarValI1 VarValI2 Label		if (VarValI1 == VarValI2) goto Label
BRA Label		goto Label
SND VarVal {VarVal}		send_to(VarVal,{VarVal}) // blocking, CSP-style
RCV VarVal {VarVal}		recv_from(VarVal,{VarVal}) // blocking, CSP-style
SLP VarValI		sleep(VarValI)
PRN {VarVal}		print values {VarVal}, followed by a new line
RET.		return
Tag = #SIMPLESCRIPT		
VarVal = VarValI VarValS.		
VarValI = Var IntVal.		
VarValS = Var StrVal.		
Var = '\$' {Letter Digit}.		
Label = '#' {Letter Digit}.		
IntVal = [-] Digit {Digit}.		
StrVal = " {Letter Digit} ".		

Ο κώδικας ενός προγράμματος SimpleScript δίνεται σε ASCII, με κάθε εντολή σε ξεχωριστή γραμμή. Τα τμήματα μιας εντολής χωρίζονται με έναν ή περισσότερους κενούς χαρακτήρες. Υποστηρίζονται μεταβλητές τύπου integer και string που δηλώνονται αυτόματα μέσω των εντολών που αναθέτουν τιμές σε αυτές. Το πρόγραμμα μπορεί να προσπελάσει τα ορίσματα που δέχεται μέσω προκαθορισμένων μεταβλητών \$arg<n>, όπου n ο αριθμός του ορίσματος. Κατά σύμβαση, το \$arg0 είναι ένα string με το όνομα του προγράμματος, και η μεταβλητή \$argc είναι ο αριθμός των ορισμάτων που περάστηκαν στο πρόγραμμα.

Αναπτύξτε έναν διερμηνέα SimpleScript που δέχεται το όνομα και τα ορίσματα ενός προγράμματος, και εκτελεί το πρόγραμμα εντολή προς εντολή. Σε περίπτωση συντακτικού λάθους ή ασυμβατότητας ορισμάτων μιας εντολής, πρέπει να εκτυπώνεται κατάλληλο μήνυμα και να τερματίζεται η εκτέλεση. Σε αυτό το στάδιο, οι εντολές γλώσσας SND/RCV πρέπει να ερμηνεύονται ως ports (θα υλοποιηθούν στο επόμενο στάδιο ανάπτυξης).

Στην συνέχεια, υλοποιήστε ένα περιβάλλον που υποστηρίζει την εντολή run <progname> <arg1> ... <argN> για την εκτέλεση ενός προγράμματος SimpleScript. Κάθε πρόγραμμα πρέπει να εκτελείται μέσα από ένα ξεχωριστό «νήμα». Η εναλλαγή ανάμεσα στα νήματα πρέπει να γίνεται αυτόματα, υπό τον έλεγχο του περιβάλλοντος εκτέλεσης (όπως επιθυμείτε). Το περιβάλλον πρέπει ακόμα να υποστηρίζει την εντολή list για την εκτύπωση του αριθμού και του προγράμματος που εκτελεί κάθε νήμα, και την εντολή kill <thread id> για τον τερματισμό ενός νήματος εκτέλεσης. Επιπλέον, στην εντολή γλώσσας PRN πρέπει να εκτυπώνεται (αυτόματα) και το αναγνωριστικό του νήματος που εκτελεί το πρόγραμμα.

Ομαδική εκτέλεση

Επεκτείνετε το περιβάλλον εκτέλεσης για να υποστηρίξει την ομαδική εκτέλεση προγραμμάτων SimpleScript, μέσω της εντολής `run <progname1> <arg1> ... <argN> || <progname2> <arg1> ... <argN> || ... || <progname3> <arg1> ... <argN>`. Τα νήματα κάθε ομάδας λαμβάνουν αυξανόμενα αριθμητικά αναγνωριστικά, αρχίζοντας από το 0 για το νήμα που εκτελεί το πρώτο πρόγραμμα που εμφανίζεται στην εντολή που δίνει ο χρήστης. Κάθε ομάδα νημάτων εκτέλεσης πρέπει να λαμβάνει αυτομάτως ένα μοναδικό αναγνωριστικό και να εκτελείται ανεξάρτητα από τις υπόλοιπες. Μια ομάδα τερματίζει την εκτέλεση της όταν τερματιστεί και το τελευταίο νήμα της ή αν προκύψει πρόβλημα στην ερμηνεία/εκτέλεση ενός των νημάτων της (οπότε τερματίζονται αυτόματα και όλα τα υπόλοιπα νήματα της ομάδας).

Σε αυτό το στάδιο πρέπει να υλοποιηθούν οι εντολές γλώσσας SND/RCV στο πνεύμα του μοντέλου μηνυμάτων CSP, όπου το πρώτο όρισμα είναι κατά σύμβαση το αναγνωριστικό του νήματος στο οποίο στέλνεται το μήνυμα / από το οποίο παραλαμβάνεται το μήνυμα, και τα υπόλοιπα ορίσματα είναι τα περιεχόμενα του μηνύματος.

Επίσης, στην εντολή γλώσσας PRN πρέπει πλέον να εκτυπώνεται και το αναγνωριστικό της ομάδας στην οποία ανήκει το νήμα που εκτελεί το πρόγραμμα. Ακόμα, η εντολή list πρέπει να επεκταθεί ώστε να εμφανίζει και το αναγνωριστικό της ομάδας μαζί με το αναγνωριστικό του κάθε νήματος, ενώ η kill πρέπει πλέον να δέχεται ως όρισμα το αναγνωριστικό της ομάδας (όχι το αναγνωριστικό ενός νήματος) και να οδηγεί στον τερματισμό όλων των νημάτων της ομάδας.

Κατανεμημένη εκτέλεση με μετανάστευση

Επεκτείνετε το περιβάλλον εκτέλεσης έτσι ώστε να υποστηρίζεται η μετανάστευση νημάτων κατά την διάρκεια της εκτέλεσης (live migration) σε ένα άλλο περιβάλλον εκτέλεσης, μέσω της εντολής `migrate <grp id> <thread id> <ip addr> <port>`. Το περιβάλλον αφετηρίας πρέπει να σώζει την κατάσταση του νήματος και να την μεταφέρει (αξιόπιστα) μαζί με τον κώδικα του νήματος στο περιβάλλον προορισμού, το οποίο πρέπει να αποκαθιστά την κατάσταση εκτέλεσης του νήματος έτσι ώστε το νήμα να συνεχίζει την εκτέλεση του από το σημείο που σταμάτησε. Ο αριθμός της ομάδας/νήματος παραμένουν ως έχουν στο περιβάλλον προορισμού (τα αναγνωριστικά των ομάδων πρέπει να είναι μοναδικά σε καθολικό επίπεδο).

Η υλοποίηση των εντολών γλώσσας SND/RCV πρέπει να επεκταθεί κατάλληλα έτσι ώστε να παρέχεται διαφάνεια επικοινωνίας στην περίπτωση που τα εμπλεκόμενα νήματα βρίσκονται σε διαφορετικό περιβάλλον εκτέλεσης ή υπό μετανάστευση.

Αυτόματη εξισορρόπηση φορτίου

Επεκτείνετε την υλοποίηση έτσι ώστε τα νήματα να μοιράζονται (περίπου) ισομερώς στα υφιστάμενα περιβάλλοντα εκτέλεσης. Υλοποιήστε ένα πρωτόκολλο έτσι ώστε τα περιβάλλοντα εκτέλεσης να ανακαλύπτουν το ένα το άλλο και να ανταλλάσσουν πληροφορία για τον αριθμό των νημάτων που έχει το καθένα, και έναν απλό αλγόριθμο για την αυτόματη λήψη αποφάσεων μετανάστευσης νημάτων (χωρίς ο χρήστης να δίνει εντολή για αυτό). Πρέπει να αποφεύγονται άσκοπες παλινδρομήσεις, όπου ένα ή περισσότερα νήματα μεταναστεύουν διαρκώς χωρίς αυτό να βελτιώνει επί της ουσίας την κατανομή των νημάτων.

Επίσης, το περιβάλλον εκτέλεσης πρέπει να υποστηρίξει την εντολή shutdown για τον ομαλό τερματισμό του, με αυτόματη μετανάστευση των νημάτων σε άλλα διαθέσιμα περιβάλλοντα εκτέλεσης. Αν στο περιβάλλον εκτέλεσης τρέχουν ακόμα κάποια νήματα και δεν υπάρχει άλλο διαθέσιμο περιβάλλον εκτέλεσης, η εντολή πρέπει να αποτυγχάνει.

Προαιρετικό: Εντοπισμός επικοινωνιακού αδιεξόδου

Σχεδιάστε και υλοποιήστε ένα μηχανισμό για τον εντοπισμό αδιεξόδων σε επίπεδο επικοινωνίας – όταν δηλαδή όλα τα νήματα μιας ομάδας έχουν μπλοκάρει στις εντολές γλώσσας SND/RCV χωρίς να υπάρχει περίπτωση προόδου.

Χρησιμοποιήστε όποια γλώσσα προγραμματισμού επιθυμείτε. Ακολουθήστε τις οδηγίες παράδοσης (στο eclass). Εκτός από τον κώδικα, πρέπει να ετοιμάσετε μια παρουσίαση που θα περιγράφει την υλοποίηση, τα πειράματά σας και τα αποτελέσματα.

Φροντιστήριο/συζήτηση: Πέμπτη 27 Μαΐου 2021 Ημερομηνία παράδοσης: Σάββατο 12 Ιουνίου 2021, 22:00
