

CST282: Μεταγλωττιστές II

Διδάσκων: Νικόλαος Καββαδίας (Λέκτορας, Π.Δ. 407/80)
nkavn@uop.gr

Εξεταστική Φεβρουαρίου 2010

Ημερομηνία εξέτασης: Τρίτη 16 Φεβρουαρίου 2010
Ώρα εξέτασης: 13:30-16:30
Ακαδημαϊκό έτος: 2009-2010

Θέματα

1) Βασικές γνώσεις.

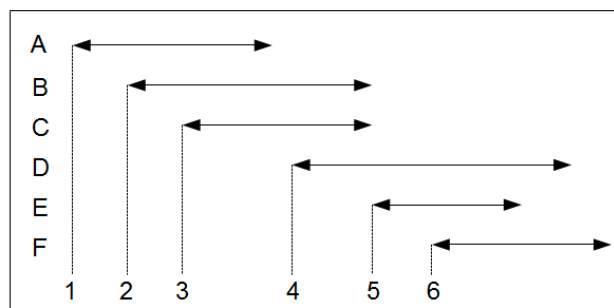
- α) Ποια τα πλεονεκτήματα της χρήσης ενδιάμεσης αναπαράστασης (IR) στο σχεδιασμό ενός επαναστοχεύσιμου μεταγλωττιστή; Να δοθεί αριθμητικό παράδειγμα για την περίπτωση μεταγλωττιστή ο οποίος δέχεται τις πηγαίες γλώσσες ANSI C και Fortran και παράγει κώδικα συμβολομεταφραστή για τις αρχιτεκτονικές x86, MIPS, ARM και PowerPC.
- β) Τι είναι βασικό μπλοκ; Δώστε ένα παράδειγμα βασικού μπλοκ (μέχρι 5 εντολές) με κώδικα τριών διευθύνσεων (TAC).

2) Ενδιάμεση αναπαράσταση και καταμερισμός καταχωρητών.

- α) Ο παρακάτω κώδικας ANSI C περιγράφει έναν αλγόριθμο παραγοντοποίησης του ακέραιου n σε γινόμενο πρώτων αριθμών (prime factorization). Πρώτος αριθμός είναι αυτός που διαιρείται ακριβώς μόνο με τον εαυτό του και τη μονάδα. Να δοθεί ο γράφος ροής ελέγχου (CFG) σε non-SSA και σε SSA μορφή για τον αλγόριθμο.

```
i = 2;
while (i <= n) {
  while ((n % i) == 0) {
    n = n / i;
    printf("%d * ", i);
  }
  i = i + 1;
}
```

- β) Να πραγματοποιηθεί καταμερισμός καταχωρητών με τον αλγόριθμο της γραμμικής σύρωσης για τους απεικονιζόμενους χρόνους ζωής έξι προσωρινών μεταβλητών (A-F). Ο αριθμός των διαθέσιμων φυσικών καταχωρητών είναι $R = 3$.



3) Βελτιστοποιήσεις.

- α) Εφαρμόστε διαδοχικά δίπλωση σταθεράς, διάδοση σταθεράς, αλγεβρικές απλοποιήσεις και εξουδετέρωση κοινής υποεκφράσεως στο παρακάτω τμήμα κώδικα.

```
if (k == 0) {  
    a = 11 + 1;  
    b = a;  
    c = (b + e) * 1024;  
    d = e + b;  
} else {  
    x = 9 * a + c / 2;  
}
```

- β) Να εφαρμοστεί πλακόστρωση βρόχων (loop tiling) για μέγεθος πλακιδίου ίσο με 16. Οι πίνακες a, b έχουν από n στοιχεία.

```
for (i = 0; i < n-1; i++) {  
    b[i] += (a[i] + a[i+1])/2;  
}
```

4) Χρονοπρογραμματισμός κώδικα.

- α) Τι είναι η εξάρτηση εντολής και τι είναι εξάρτηση δεδομένων; Να αναφερθούν τα είδη εξάρτησης δεδομένων.
- β) Έστω η υποθετική αρχιτεκτονική RISC με τις εντολές ADD, MUL, LOAD και STORE οι οποίες περιγράφονται στον παρακάτω πίνακα.

Διαμόρφωση	Συμπεριφορά	Κύκλοι μηχανής
ADD R1, R2, R3	R1 := R2 + R3	1
MUL R1, R2, R3	R1 := R2 * R3	2
LOAD R1, imm(R2)	R1 := MEM(R2 + imm)	3
STORE imm(R2), R1	MEM(R2 + imm) := R1	3

Να πραγματοποιηθεί ο χρονοπρογραμματισμός του παρακάτω τμήματος κώδικα ώστε ο συνολικός χρόνος εκτέλεσης σε αρχιτεκτονική με μία ALU και μία μονάδα φόρτωσης-αποθήκευσης (LOAD/STORE) να μειωθεί από τους 20 στους 13. Δίνονται οι κύκλοι εκτέλεσης για κάθε εντολή στο αρχικό χρονοπρόγραμμα.

```
1-3:  LOAD  R1, 0(R30)  
4-4:  ADD   R1, R1, R1  
5-7:  LOAD  R2, 1(R30)  
8-9:  MUL   R1, R1, R2  
9-11: LOAD  R3, 2(R30)  
12-13: MUL  R1, R1, R3  
13-15: LOAD R2, 3(R30)  
16-17: MUL  R1, R1, R2  
18-20: STORE 4(R30), R1
```