

Πανεπιστήμιο Θεσσαλίας
Τμήμα Πληροφορικής και Τηλεπικοινωνιών

Μεταγλωττιστές

Εργασία Χειμερινού Εξαμήνου 2024-2025

Σκοπός της εργασίας εξαμήνου είναι η κατασκευή ενός μεταγλωττιστή για μια δομημένη γλώσσα προγραμματισμού υψηλού επιπέδου.

Η γλώσσα αυτή θα είναι μια από τις FORT, PASC και CPP, οι οποίες είναι απλοποιημένες παραλλαγές των FORTRAN, PASCAL και C++, αντίστοιχα.

Η γλώσσα υλοποίησης του μεταγλωττιστή είναι η C.

Θα ακολουθήσετε 3 στάδια κατασκευής, τα εξής:

1. Κατασκευή λεκτικού και συντακτικού αναλυτή.
2. Κατασκευή ενδιάμεσου κώδικα σε μορφή αφηρημένου συντακτικού δέντρου και σημασιολογικού αναλυτή.
3. Παραγωγή τελικού κώδικα σε συμβολική γλώσσα MIPS ή IA32 και (προαιρετικά) βελτιστοποίηση με βάση το αφηρημένο συντακτικό δέντρο.

Παράλληλα θα αναπτύξετε τον πίνακα συμβόλων του μεταγλωττιστή, συλλέγοντας πληροφορίες στις φάσεις της ανάλυσης, ώστε να τις χρησιμοποιήσετε για την παραγωγή του ενδιάμεσου κώδικα.

Περισσότερες λεπτομέρειες για καθένα από τα πιο πάνω στάδια δίνονται παρακάτω.

Για την ολοκλήρωση όσο το δυνατόν μεγαλύτερου μέρους της εργασίας μέχρι το τέλος του εξαμήνου, θα συνεργαστείτε σε ομάδες μέχρι 3 ατόμων.

Η επιλογή της γλώσσας που θα μεταφράσετε θα βρεθεί από το υπόλοιπο της διαίρεσης του αθροίσματος των ΑΕΜ των μελών της ομάδας με το 3. Αν αυτό είναι 0, η γλώσσα θα είναι η FORT. Αν είναι 1, η γλώσσα θα είναι η PASC. Αν τέλος είναι 2, η γλώσσα θα είναι η CPP.

Η περιγραφή των 3 γλωσσών θα βρίσκεται στα αντίστοιχα αρχεία:

fort.pdf
pasc.pdf
cpp.pdf

στους σχετικούς φακέλους στα Έγγραφα του μαθήματος στην πλατφόρμα eclass, μαζί με προγράμματα δοκιμής.

Στάδιο 1

Να κατασκευάσετε το Λεκτικό και το Συντακτικό Αναλυτή (ΛΑ, ΣΑ) για τη γλώσσα της εργασίας σας. Για κάποιο τυχαίο πρόγραμμα εισόδου, ο ΣΑ καλείται μία μοναδική φορά, για να αποφασίσει αν το πρόγραμμα είναι συντακτικά ορθό, ενώ ο ΛΑ καλείται επαναληπτικά από το ΣΑ, κάθε φορά που ο δεύτερος χρειάζεται να προχωρήσει σε επόμενη Λεκτική Μονάδα (ΛΜ) του προγράμματος.

A. Πριν προχωρήσετε στην κατασκευή του ΛΑ, θα πρέπει να σχεδιάσετε το Διάγραμμα Μετάβασης (ΔΜ) της γλώσσας της εργασίας σας σε ενιαία μορφή. Τοποθετήστε πρώτα τις λέξεις-κλειδιά, στη συνέχεια προσθέστε τις σταθερές και τα αναγνωριστικά, και τέλος συμπληρώστε τους τελεστές και τα διαχωριστικά σύμβολα. Να κατασκευάσετε έναν πίνακα με όλες τις τελικές καταστάσεις του ΔΜ, όπου να σημειώσετε τις ΛΜ που επιστρέφονται, καθώς και την οπισθοδρόμηση που πρέπει να γίνεται σε κάθε τελική κατάσταση.

Προχωρήστε τώρα στην υλοποίηση του ΛΑ, ο οποίος θα πρέπει:

1. Να επιστρέφει τον κωδικό της ΛΜ που βρίσκει κάθε φορά που καλείται.
2. Να επιστρέφει τα ονόματα των αναγνωριστικών που συναντά, ή εναλλακτικά να εισάγει στον Πίνακα Συμβόλων (ΠΣ) τα αναγνωριστικά που συναντά, επιστρέφοντας το δείκτη στον ΠΣ που αντιστοιχεί στο τρέχον αναγνωριστικό.
3. Να επιστρέφει τις τιμές των σταθερών που συναντά. Για μη αριθμητικές σταθερές να επιστρέφει τη συμβολοσειρά της λέξης που αναγνώρισε.
4. Να χειρίζεται σφάλματα λεκτικής ανάλυσης με εκτύπωση μηνύματος που να περιλαμβάνει τον αριθμό γραμμής και την ίδια τη γραμμή του αρχείου εισόδου στην οποία σημειώθηκε το σφάλμα. Να προσπαθεί να συνεχίσει σβήνοντας το λανθασμένο χαρακτήρα, και αν αυτό δεν είναι δυνατό, τότε να τερματίζει τη λειτουργία του.
5. Για τους σκοπούς της παρούσας άσκησης να εκτυπώνει κατάλληλα όλες τις τιμές που επιστρέφει, πέρα από τυχόν μηνύματα λάθους.

Για την κατασκευή του ΛΑ μπορείτε να χρησιμοποιήσετε είτε τη μέθοδο του απ' ευθείας προγραμματισμού του ΔΜ που σχεδιάσατε, είτε το μετα-εργαλείο flex. Η κωδικοποίηση των ΛΜ να γίνει με το συμβολισμό που δίνεται στην περιγραφή της γλώσσας που μεταφράζετε. Για παράδειγμα, το σύμβολο `ID` να χρησιμοποιηθεί για την κωδικοποίηση των αναγνωριστικών. Σε περίπτωση που χρησιμοποιήσετε το flex, θα πρέπει οι πληροφορίες που ο ΛΑ επιστρέφει πέρα από τον κωδικό της ΛΜ, να επιστρέφονται μέσω της μεταβλητής `yylval`.

Για να γίνει δεκτή η παράδοση του ΛΑ αν χρησιμοποιήσετε το flex, θα πρέπει ο πιο πάνω πίνακας τελικών καταστάσεων του ΔΜ να συμπληρωθεί με την αντιστοίχιση κάθε αριθμού κατάστασης του δικού σας ΔΜ με τον αριθμό κατάστασης του ΔΜ που κατασκευάζει το flex. Για το σκοπό αυτό, θα πρέπει να ανοίξετε το αρχείο που παράγει το flex και να εντοπίσετε τις καταστάσεις του ΔΜ που αναγνωρίζουν τις ΛΜ της γλώσσας. Επειδή το αρχείο αυτό παράγεται αυτόματα, φροντίστε να το μετονομάσετε σε άλλο κατάλληλο όνομα, για να μην μπερδευτείτε και το χάσετε σε κάποια επόμενη εκτέλεση του flex. Το συγκεκριμένο αρχείο που χρησιμοποιήσατε θα πρέπει να παραδοθεί μαζί με το υπόλοιπο υλικό του ΛΑ.

B. Ο ΣΑ θα πρέπει:

1. Να αναγνωρίζει ή να απορρίπτει το πρόγραμμα με βάση τη γραμματική της γλώσσας. Πιο συγκεκριμένα: Για να κατασκευαστεί ένας ντετερμινιστικός ΣΑ για τη γλώσσα της εργασίας σας, πρέπει η γραμματική να μην είναι διφορούμενη. Για το σκοπό αυτό πρέπει να μετατρέψετε τη διφορούμενη γραμματική που σας δίνεται σε μη διφορούμενη με έναν από τους εξής δύο τρόπους: (α) μετασχηματίζοντας τους κανόνες της γραμματικής, ή (β) χρησιμοποιώντας ιδιότητες της γλώσσας όπως την προτεραιότητα και την προσεταιριστικότητα των τελεστών της.
2. Να χειρίζεται σφάλματα συντακτικής ανάλυσης με εκτύπωση κατάλληλων μηνυμάτων που να περιλαμβάνουν τον αριθμό γραμμής και την ίδια τη γραμμή του αρχείου εισόδου στην οποία σημειώθηκε κάθε σφάλμα. Ειδικότερα: Για ανάνηψη από κάποιο σφάλμα να χρησιμοποιηθεί η μέθοδος πανικού σε συνδυασμό με τη στρατηγική των κανόνων παραγωγής σφαλμάτων για τα πιο κοινά σφάλματα, όπως η απουσία διαχωριστικών. Μετά το πρώτο σφάλμα να συνεχίζεται η ανάλυση μέχρι έναν αριθμό 5 το πολύ σφαλμάτων.
3. Να εισάγει στον ΠΣ τα αναγνωριστικά του προγράμματος, αν δεν έχουν εισαχθεί από το ΛΑ. Ακόμα, θα πρέπει να διαχειρίζεται τις εμβέλεις του προγράμματος στον ΠΣ, σύμφωνα με την περιγραφή της γλώσσας που μεταφράζετε.

4. Για τους σκοπούς της παρούσας άσκησης να εκτυπώνει με κατάλληλο τρόπο το περιεχόμενο του ΠΣ στο τέλος κάθε εμβέλειας.

Η χρήση του μετα-εργαλείου bison είναι αναγκαία, τόσο για την κατασκευή του ΣΑ, όπου μπορούμε να χειριστούμε διαφορούμενες γραμματικές χωρίς την ανάγκη μετασχηματισμού, όσο και για την ενσωμάτωση σε αυτόν του σημασιολογικού αναλυτή. Με το bison μπορούμε επίσης να υλοποιήσουμε ευκολότερα τη μέθοδο πανικού για ανάνηψη από συντακτικά σφάλματα.

Για να γίνει δεκτή η παράδοση του ΣΑ, θα πρέπει να μελετήσετε το αρχείο ανάλυσης της γραμματικής που παράγει το bison (με επιλογή “-v”), και (α) σε περίπτωση που δεν μετασχηματίσετε τη γραμματική σας σε μη διαφορούμενη, θα πρέπει να βρείτε και να υποδείξετε τις καταστάσεις στις οποίες εμφανίζονται συγκρούσεις, καθώς και τις καταστάσεις όπου οι συγκρούσεις του ΣΑ επιλύονται με βάση την προτεραιότητα και προσεταιριστικότητα των τελεστών, δίνοντας και τα εμπλεκόμενα στοιχεία του ΣΑ, ενώ (β) σε περίπτωση που μετασχηματίσετε τη γραμματική σας σε μη διαφορούμενη, θα πρέπει να βρείτε και να υποδείξετε τις καταστάσεις στις οποίες εμφανίζονται συγκρούσεις *πριν το μετασχηματισμό*, δίνοντας και τα εμπλεκόμενα στοιχεία του ΣΑ, καθώς και τις νέες καταστάσεις που προκύπτουν *μετά το μετασχηματισμό*, όπου οι συγκρούσεις θα έχουν επιλυθεί.

Όσο αφορά τον ΠΣ, μπορείτε να τον σχεδιάσετε με όποια δομή θέλετε. Προς το παρόν η μόνη πληροφορία που αποθηκεύεται στον ΠΣ για κάποιο αναγνωριστικό είναι το όνομά του και το βάθος φωλιάσματος της εμβέλειάς του.

Για επαλήθευση τόσο του ΛΑ όσο και του ΣΑ χρησιμοποιήστε τα αρχεία εισόδου που επισυνάπτονται του αρχείου περιγραφής της γλώσσας της εργασίας σας.

Στάδιο 2

Να κατασκευάσετε το Σημασιολογικό Αναλυτή για τη γλώσσα της εργασίας σας, μέσα από το Συντακτικό Αναλυτή (ΣΑ) που θα πρέπει να έχετε ήδη κατασκευάσει. Ο συνολικός Συντακτικός και Σημασιολογικός Αναλυτής (ΣΣΑ) πρέπει να καλείται μια φορά για δοθέν πρόγραμμα, και παράλληλα με τη συντακτική ανάλυση, θα πρέπει:

1. Να αποτιμά τα κατηγορήματα των συμβόλων της γραμματικής. Με βάση την περιγραφή της γλώσσας, να ορίσετε ένα σύνολο κατηγορημάτων για κάθε σύμβολο, έτσι ώστε να συγκεντρώνετε όποια πληροφορία χρειάζεστε για τη συνέχεια της μετάφρασης. Σχεδιάστε έτσι τον αναλυτή σας, ώστε να μπορείτε να προσθέτετε κατηγορήματα όταν διαπιστώνετε ότι είναι απαραίτητα. Μπορείτε να χρησιμοποιείτε καθολικές μεταβλητές για αποθήκευση πληροφορίας, προσπαθήστε όμως να χρησιμοποιείτε κατηγορήματα όσο μπορείτε, επειδή αυτά εισάγονται στη στοίβα αποτίμησης του ΣΑ κι επομένως διατηρούν την τιμή τους σε περιπτώσεις πολλαπλών εμφανίσεων του ίδιου συμβόλου ή σε περιπτώσεις φωλιασμένων δομών της γλώσσας.
2. Να προσθέτει στον ΠΣ πληροφορίες για τα αναγνωριστικά που έχουν εισαχθεί από το ΛΑ ή το ΣΑ. Έτσι:
 - (α) Να κωδικοποιούνται στον ΠΣ οι τύποι και τα συνώνυμα τύπων, όπου αυτά υπάρχουν.
 - (β) Να προσδιορίζονται οι τύποι των μεταβλητών και των συναρτήσεων κατά τη δήλωσή τους.
 - (γ) Να αποδίδονται τιμές σε ονόματα σταθερών ή πιθανές αρχικές τιμές σε ονόματα μεταβλητών.
 - (δ) Να προσδιορίζονται ο αριθμός, οι τύποι, ο τρόπος περάσματος και τα ονόματα των παραμέτρων των υποπρογραμμάτων που δηλώνονται στο πρόγραμμα.
3. Αν η γλώσσα περιλαμβάνει εντολές *goto*, να τις συνδέει με τις αντίστοιχες ετικέτες. Για το σκοπό αυτό οι ετικέτες μπορούν να εισάγονται στον ΠΣ και να αντιμετωπίζονται σαν τα υπόλοιπα αναγνωριστικά.

4. Να κατασκευάζει το Αφηρημένο Συντακτικό Δέντρο (ΑΣΔ) του προγράμματος. Κάθε κόμβος κατασκευάζεται από τον κώδικα ενέργειας που συνοδεύει την αναγνώριση κάποιου συντακτικού κανόνα της γραμματικής. Ειδικότερα:
 - (α) Για τις εκφράσεις συνιστάται η χρήση τύπου δέντρου με μεταβλητό αριθμό παιδιών – ανάλογα με τον τελεστή κάθε κόμβου, ενώ για τις εντολές αρκεί η χρήση γραμμικής λίστας με δυνατότητα φωλιάσματος – για υποστήριξη σύνθετων και δομημένων εντολών. Για παράδειγμα, ο κόμβος μιας εντολής διακλάδωσης συνδέεται με τους κόμβους της προηγούμενης και της επόμενης εντολής, αλλά περιέχει και ένα ή δύο δείκτες σε κόμβους εντολών που αντιστοιχούν στις κατευθύνσεις της διακλάδωσης. Ας σημειωθεί ότι ο ίδιος κόμβος περιέχει και ένα δείκτη σε ρίζα δέντρου έκφρασης. Γενικά, οι περισσότεροι κόμβοι εντολών περιέχουν δείκτες σε δέντρα έκφρασης.
 - (β) Η δομή των κόμβων του ΑΣΔ πρέπει να είναι τέτοια, ώστε να περικλείει όλη την πληροφορία που παράγεται από τη σημασιολογική ανάλυση των ίδιων και είναι απαραίτητη για τη σημασιολογική ανάλυση των γειτονικών κόμβων. Έτσι, ένας κόμβος έκφρασης πρέπει να αποθηκεύει πληροφορία για τον τύπο του, η οποία παράγεται από τη σημασιολογική ανάλυσή του και χρησιμοποιείται για τη σημασιολογική ανάλυση των γονέων του.
 - (γ) Το δέντρο πρέπει να είναι όσο γίνεται πιο αφηρημένο, ώστε να είναι μικρό και να περιλαμβάνει μόνο τους αναγκαίους κόμβους για τη μετέπειτα ανάλυση του προγράμματος. Για παράδειγμα, κόμβοι για την αποθήκευση διαχωριστικών – πχ παρενθέσεων – είναι σκόπιμο να παραλείπονται, ενώ οι εκφράσεις είναι προτιμότερο να αποθηκεύονται έτσι, ώστε σε κάθε εσωτερικό κόμβο τους να αποθηκεύεται ο αντίστοιχος τελεστής, αντί κάποιου μη τερματικού συμβόλου της γραμματικής.Να έχετε υπ’ όψη ότι ο τύπος του κόμβου του ΑΣΔ θα παίζει σημαντικό ρόλο στη βελτιστοποίηση και την παραγωγή τελικού κώδικα, γι’ αυτό θα πρέπει να είναι εύκολος στο χειρισμό και να μπορεί να τροποποιείται εύκολα. Καλό θα είναι να διαχωρίσετε τον κώδικα C που διαχειρίζεται το ΑΣΔ σε κάποιο αρχείο έξω από το κυρίως αρχείο του ΣΣΑ.
5. Να δεσμεύει θέσεις στο χώρο δεδομένων (ΧΔ) για τις μεταβλητές του προγράμματος, δημιουργώντας πίνακες δέσμευσης για κάθε εμβέλεια που μεταφράζεται. Για τη γλώσσα της εργασίας σας καθορίστε τις κατηγορίες αποθήκευσης που υποστηρίζονται, το δέσιμο των μεταβλητών, το φώλιασμα των εμβελειών, και συμπληρώστε στον ΠΣ τα στοιχεία που θεωρείτε απαραίτητα για την παραγωγή τελικού κώδικα της επίλυσης αναφοράς κάθε μεταβλητής.
6. Να εκτελεί τον έλεγχο των σημασιολογικών κανόνων της γλώσσας. Πιο συγκεκριμένα:
 - (α) Να ελέγχει την ορθότητα όλων των δηλώσεων. Έτσι για παράδειγμα δεν πρέπει να υπάρχουν πολλαπλές δηλώσεις για το ίδιο αναγνωριστικό στην ίδια εμβέλεια. Ή ακόμα οι δηλώσεις συναρτήσεων πρέπει να ταιριάζουν με πιθανή προηγούμενη δήλωση των πρωτοτύπων τους, ο τύπος δήλωσης μιας σταθεράς πρέπει να ταιριάζει με την τιμή που της αποδίδεται, και ο τύπος μιας μεταβλητής πρέπει να ταιριάζει με την πιθανή αρχική της τιμή.
 - (β) Να ελέγχει τη χρήση όλων των αναγνωριστικών σε σχέση με τη δήλωσή τους. Η χρήση αυτών πρέπει να γίνεται με τρόπο συμβατό με την αντίστοιχη δήλωση και στην εμβέλεια στην οποία έχουν δηλωθεί.
 - (γ) Να ελέγχει τους τύπους των επιμέρους εκφράσεων σε μεγαλύτερες εκφράσεις. Αν αυτό κριθεί αναγκαίο, κόμβοι μετατροπής συμβατών τύπων πρέπει να εισάγονται στο ΑΣΔ.
 - (δ) Να ελέγχει τη σημασιολογική ορθότητα των εντολών του προγράμματος. Για παράδειγμα, εδώ ελέγχονται οι τύποι στις αναθέσεις και προστίθενται τελικοί κόμβοι μετατροπής τύπων από το δεξί στο αριστερό μέλος μιας ανάθεσης, αν αυτό είναι απαραίτητο. Επίσης ελέγχονται οι κλήσεις υποπρογραμμάτων για την ορθότητα του αριθμού και του τύπου των παραμέτρων. Ή ακόμα ελέγχονται οι τύποι παραστάσεων που υπεισέρχονται σε συγκεκριμένες εντολές, όπως εντολές διακλάδωσης και εντολές βρόχων. Τέλος, εδώ ελέγχονται τα άλματα εντολών *goto*, και γενικά κάθε είδους εντολές ελέγχου ροής σε καθεμία από τις τρεις γλώσσες της εργασίας σας.
7. Σε περίπτωση σφάλματος συντακτικής ανάλυσης να σταματά την παραγωγή κώδικα. Σε περίπτωση σημασιολογικού σφάλματος, ο ΣΣΑ να τερματίζει άμεσα τη λειτουργία του.
8. Για τους σκοπούς της παρούσας άσκησης να εκτυπώνει με κατάλληλο τρόπο το ΑΣΔ και το ΧΔ στο τέλος κάθε μονάδας του προγράμματος εισόδου.

Η σημασιολογική ανάλυση γίνεται μέσα από τις ενέργειες του μετα-εργαλείου Bison. Ας σημειωθεί ότι επειδή το Bison υποστηρίζει ένα μόνο κατηγορήμα ανά σύμβολο της γραμματικής, η υλοποίηση πολλαπλών κατηγορημάτων θα πρέπει να γίνει με το δομημένο τύπο `struct` της C. Τις πιο πολλές φορές, ένα σύμβολο της γραμματικής αποδίδει έναν κόμβο του ΑΣΔ, ο οποίος επιστρέφεται μέσα από το κατηγορήμα του συμβόλου στον πατρικό κόμβο, οπότε τα όποια επιπλέον κατηγορήματα του συμβόλου μπορούν να ενσωματωθούν στη δομή του αντίστοιχου κόμβου.

Στάδιο 3

Να κατασκευάσετε το Γεννήτορα Τελικού Κώδικα (ΓΤΚ) για τη γλώσσα της εργασίας σας, ο οποίος πρέπει να καλείται μια φορά μετά την παραγωγή ενδιάμεσου κώδικα, και να παράγει τον τελικό κώδικα που αντιστοιχεί σε τυχόν πρόγραμμα εισόδου.

Σαν τελική γλώσσα να χρησιμοποιήσετε τη συμβολική γλώσσα του συνόλου εντολών MIPS που διδαχθήκατε στο μάθημα της Εισαγωγής στους H/Y. Εναλλακτικά μπορείτε να χρησιμοποιήσετε τη συμβολική γλώσσα του συνόλου εντολών IA32.

Ο ΓΤΚ πρέπει να προβαίνει στις ακόλουθες ενέργειες:

1. Ανάλογα με το μέγεθος αναπαράστασης σε bytes των βασικών τύπων της γλώσσας σας που υποστηρίζονται από την τελική γλώσσα, να προσθέτει στους πίνακες δέσμευσης θέσεων των χώρων δεδομένων του προγράμματος τις ακριβείς τιμές μετατόπισης κάθε θέσης από την αρχή του αντίστοιχου χώρου.
2. Να μετατρέπει κάθε κόμβο τελεστή αναφοράς σε στοιχείο πίνακα ή σε πεδίο εγγραφής/κλάσης του ενδιάμεσου κώδικα, σε έναν ή περισσότερους κόμβους με τις ακριβείς πράξεις που πρέπει να εκτελεστούν για τον υπολογισμό των διευθύνσεων των αντίστοιχων στοιχείων ή πεδίων.
3. Να μετατρέπει κάθε κόμβο κλήσης υποπρογράμματος του ενδιάμεσου κώδικα, σε έναν ή περισσότερους κόμβους που να αντιστοιχούν στις ακριβείς ενέργειες που συνοδεύουν μια κλήση. Δηλαδή (α) αποθήκευση των τιμών των πραγματικών παραμέτρων στη στοίβα, στις προκαθορισμένες θέσεις που αντιστοιχούν στις τυπικές παραμέτρους του υποπρογράμματος, (β) άλμα με σύνδεση στον κώδικα του υποπρογράμματος, και (γ) για συναρτήσεις, την ανάγνωση της τιμής του αποτελέσματος από την αντίστοιχη θέση στη στοίβα μετά την κλήση.
4. Για κάθε βαθμωτή μεταβλητή, στοιχείο πίνακα ή πεδίο εγγραφής/κλάσης που αποτελεί τιμή δεξιάς προσπέλασης στον ενδιάμεσο κώδικα, να εισάγει έναν κόμβο αποδεικτοδότησης, δηλαδή φόρτωσης από τη διεύθυνση της μεταβλητής ή του στοιχείου ή πεδίου αυτού.
5. Να υποστηρίζει τις γνωστές συναρτήσεις εισόδου/εξόδου `printf` και `scanf` της C, με αντίστοιχα ονόματα `“_printf”` και `“_scanf”`. Έτσι, θα πρέπει να μετατρέπει τα ορίσματα των εντολών εισόδου/εξόδου που υπάρχουν στον ενδιάμεσο κώδικα σε παραμέτρους αυτών των συναρτήσεων. Ως πρώτη παράμετρο θα πρέπει να κατασκευάζει τη συμβολοσειρά περιγραφής των πεδίων εισόδου/εξόδου της εντολής, την οποία θα πρέπει να αποθηκεύει στο χώρο δεδομένων του προγράμματος, ώστε η διεύθυνσή της να γίνεται η πρώτη πραγματική παράμετρος στην κλήση της αντίστοιχης συνάρτησης.
6. Να αποδίδει προσωρινές μεταβλητές σε εκείνους τους κόμβους του ενδιάμεσου κώδικα που αντιστοιχούν σε εφαρμογή κάποιου τελεστή ή φορτώνουν κάποια τιμή από τη μνήμη. Η απόδοση των προσωρινών μεταβλητών γίνεται με αρίθμηση των αντίστοιχων κόμβων με τη σειρά αποτίμησής τους, όπως αυτή προκύπτει μετά από ανάλυση ροής ελέγχου του προγράμματος. Αν δεν έχει γίνει προηγούμενη βελτιστοποίηση, η σειρά αυτή καθορίζεται απλά από μια διαπέραση κάθε δέντρου έκφρασης του ενδιάμεσου κώδικα.
7. Να δεσμεύει καταχωρητές για την αποτίμηση κάθε έκφρασης του ενδιάμεσου κώδικα, κατασκευάζοντας με βάση την παραπάνω αρίθμηση και χρωματίζοντας το γράφο αλληλεπιδράσεων. Αν δεν έχει προηγηθεί βελτιστοποίηση που να έχει συνδέσει τα δέντρα εκφράσεων μεταξύ τους, η δέσμευση καταχωρητών γίνεται ανεξάρτητα σε κάθε έκφραση.

8. Να εισάγει κόμβους διάχυσης για αποθήκευση και επανάκτηση των τιμών των προσωρινών μεταβλητών που δε μπορούν να αντιστοιχηθούν σε καταχωρητές. Οι θέσεις αυτών θα βρίσκονται στον τοπικό χώρο δεδομένων, και μετά από τις υπόλοιπες τοπικές μεταβλητές της μονάδας, οπότε το μέγεθος του χώρου αυτού αυξάνεται ανάλογα.
9. Να εισάγει κόμβους διάχυσης σε κάθε κόμβο κλήσης υποπρογράμματος του ενδιάμεσου κώδικα, για αποθήκευση και επανάκτηση τιμών καταχωρητών που πρέπει να διατηρούν την τιμή τους μέσα από την κλήση. Όπως και προηγουμένως, δεσμεύονται θέσεις του τοπικού χώρου δεδομένων γι' αυτό το σκοπό.
10. Να παράγει τον τελικό κώδικα, με βάση ένα σχέδιο για κάθε δυνατό κόμβο του ενδιάμεσου κώδικα – δηλαδή κάθε δυνατό κόμβο τελεστή, κόμβο προσπέλασης μνήμης και κόμβο εντολών. Ο ΓΤΚ θα πρέπει έτσι να παράγει συγκεκριμένο τελικό κώδικα για κάθε τύπο κόμβου που συναντά, καθώς διαπερνά τον ενδιάμεσο κώδικα με τη σειρά εκτέλεσης των αντίστοιχων εντολών. Ιδιαίτερη προσοχή απαιτείται σε εισόδους και εξόδους μονάδων, οπότε πρέπει να γίνεται διαχείριση της στοίβας και πιθανή διάχυση για αποθήκευση και επανάκτηση τιμών καταχωρητών που δεν πρέπει να αλλάζουν τιμή μέσα στη μονάδα. Για αναφορές στη μνήμη, να χρησιμοποιούνται οι καταχωρητές \$gr και \$sr σε δείκτες στους χώρους καθολικών/στατικών δεδομένων και στοίβας, αντίστοιχα.

Προαιρετικά, προσθέστε στο μεταγλωττιστή σας βελτιστοποίηση στο επίπεδο του ενδιάμεσου κώδικα. Πιθανές βελτιστοποιήσεις που μπορείτε να επιλέξετε είναι:

1. Αποτίμηση και διάδοση σταθερών εκφράσεων.
2. Στοιχειώδης εύρεση και απαλοιφή κοινών υποεκφράσεων
3. Εύρεση και απαλοιφή άχρηστου κώδικα.
4. Διάδοση αντιγράφου.
5. Εύρεση αναλλοίωτου κώδικα σε βρόχους και εξαγωγή εκτός βρόχου.

Ο βαθμός οποιας βελτιστοποίησης υλοποιήσετε θα προστεθεί στο βαθμό της εργασίας.