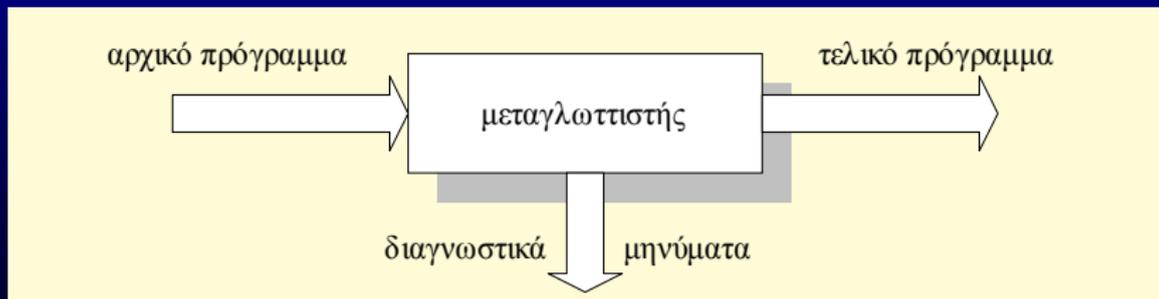


# Εισαγωγή (i)

- ▶ Γλώσσες προγραμματισμού
- ▶ Μεταγλωττιστές
- ▶ Αναγκαιότητα και ιστορική αναδρομή

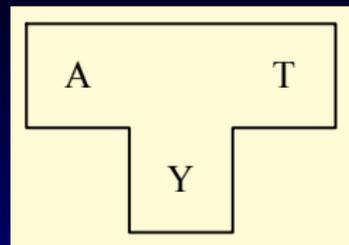
# Εισαγωγή (i)

- ▶ Γλώσσες προγραμματισμού
- ▶ Μεταγλωττιστές
- ▶ Αναγκαιότητα και ιστορική αναδρομή



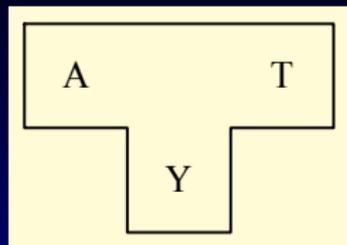
# Εισαγωγή (ii)

- ▶ Αρχική γλώσσα  $L_A$
- ▶ Τελική γλώσσα  $L_T$
- ▶ Γλώσσα υλοποίησης  $L_Y$



# Εισαγωγή (ii)

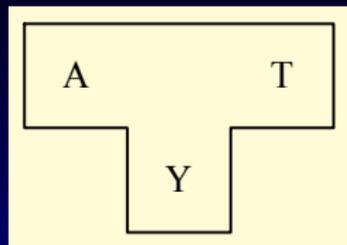
- ▶ Αρχική γλώσσα  $L_A$
- ▶ Τελική γλώσσα  $L_T$
- ▶ Γλώσσα υλοποίησης  $L_Y$
- ▶ Σημασιολογία του προγράμματος  $P$  στη γλώσσα  $L$



$$\llbracket P \rrbracket_L : \text{Inp}(P) \rightarrow \text{Out}(P)$$

# Εισαγωγή (ii)

- ▶ Αρχική γλώσσα  $L_A$
- ▶ Τελική γλώσσα  $L_T$
- ▶ Γλώσσα υλοποίησης  $L_Y$



- ▶ Σημασιολογία του προγράμματος  $P$  στη γλώσσα  $L$

$$\llbracket P \rrbracket_L : Inp(P) \rightarrow Out(P)$$

- ▶ Σημασία ενός μεταγλωττιστή  $C$

$$\llbracket C \rrbracket_{L_Y} : L_A \rightarrow L_T$$

# Εισαγωγή (iii)

- ▶ **Ορθότητα** του μεταγλωττιστή: “το μεταγλωττισμένο πρόγραμμα πρέπει να είναι **ισοδύναμο** με το αρχικό”

# Εισαγωγή (iii)

- ▶ **Ορθότητα** του μεταγλωττιστή: “το μεταγλωττισμένο πρόγραμμα πρέπει να είναι **ισοδύναμο** με το αρχικό”

$$\llbracket P \rrbracket_{L_A} = \llbracket \llbracket C \rrbracket_{L_Y}(P) \rrbracket_{L_T}$$

# Εισαγωγή (iv)

- ▶ Είδη μεταγλωττιστών:
  - ▶ Απλοί
  - ▶ Αντίστροφοι (decompilers)
  - ▶ Μετα-μεταγλωττιστές (meta-compilers)
- ▶ Ειδικές περιπτώσεις μεταγλωττιστών:
  - ▶ Προεπεξεργαστές (preprocessors)
  - ▶ Συμβολομεταφραστές (assemblers)
  - ▶ Γεννήτορες προγραμμάτων (program generators)

# Εισαγωγή (v)

## ▶ Συναφή εργαλεία

- ▶ Διερμηνείς (interpreters)
- ▶ Διαχειριστές βιβλιοθηκών (library managers)
- ▶ Συνδέτες (linkers)
- ▶ Φορτωτές (loaders)
- ▶ Εκδότες προγραμμάτων (program editors)
- ▶ Εντοπιστές σφαλμάτων (debuggers)
- ▶ Στατιστικοί αναλυτές (profilers)

# Κατασκευή μεταγλωττιστή (i)

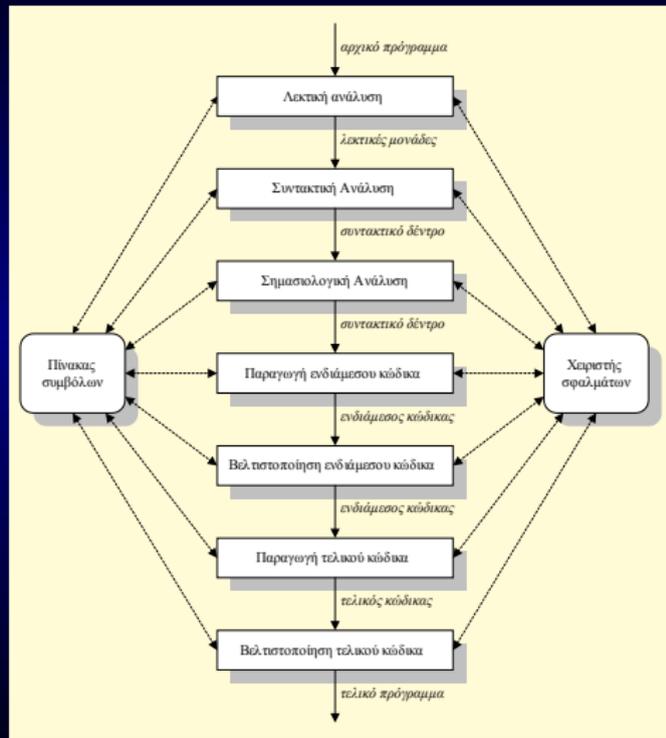
- ▶ Βασικές απαιτήσεις:
  - ▶ Να λειτουργεί σωστά
  - ▶ Να συμμορφώνεται με τις προδιαγραφές της αρχικής και της τελικής γλώσσας
  - ▶ Να μεταγλωττίζει προγράμματα κάθε μεγέθους

# Κατασκευή μεταγλωττιστή (ii)

- ▶ Επιπρόσθετες απαιτήσεις:
  - ▶ Να παράγει αποδοτικό κώδικα
  - ▶ Να έχει μικρό χρόνο μεταγλώττισης
  - ▶ Να έχει μικρές απαιτήσεις μνήμης κατά τη μεταγλώττιση
  - ▶ Να δίνει καλά διαγνωστικά μηνύματα
  - ▶ Να συνεχίζει ύστερα από λάθη
  - ▶ Να είναι μεταφέρσιμος

# Φάσεις και προϊόντα της μεταγλώττισης

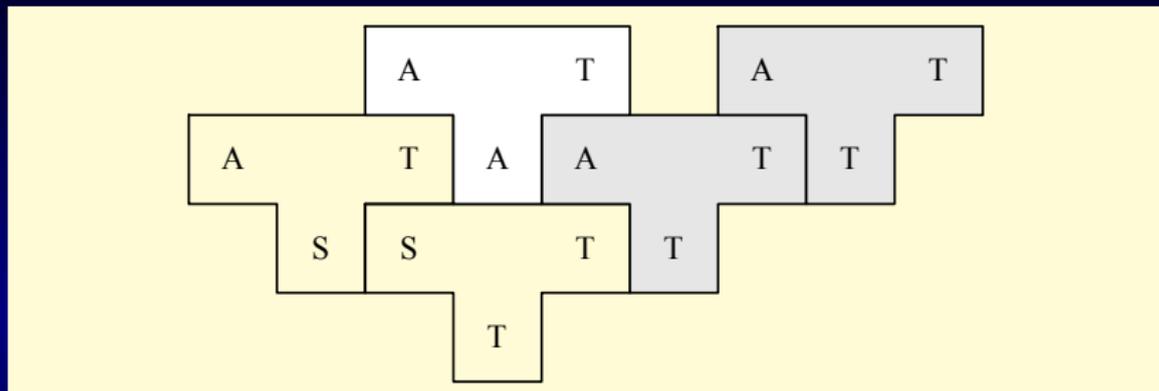
- ▶ Λεκτική ανάλυση
- ▶ Συντακτική ανάλυση
- ▶ Σημασιολογική ανάλυση
- ▶ Παραγωγή ενδιάμεσου κώδικα
- ▶ Βελτιστοποίηση
- ▶ Παραγωγή τελικού κώδικα



# Θέματα υλοποίησης

- ▶ Οργάνωση σε περάσματα
- ▶ Οργάνωση σε εμπρόσθιο και οπίσθιο τμήμα (front-end / back-end)
- ▶ Έλεγχος ορθότητας
- ▶ Είδη διαγνωστικών μηνυμάτων και ανάνηψη
  - ▶ Εσωτερικά (internal)
  - ▶ Σφάλματα (errors)
  - ▶ Προειδοποιητικά μηνύματα (warnings)
  - ▶ Απλά μηνύματα (messages)

# Εκκίνηση – bootstrapping



- ▶ **Βήμα 1:** Μεταγλωττιστής για  $S \subset A$  στην  $T$ .
- ▶ **Βήμα 2:** Μεταγλωττιστής για την  $A$  στην  $S$ .
- ▶ **Βήμα 3:** Μεταγλωττιστής για την  $A$  στην  $A$ .

# Οργάνωση σε ένα πέρασμα

