

ΤΕΚΜΗΡΙΩΣΗ

Το ζητούμενο της άσκησης αυτής είναι η υλοποίηση μιας γεννήτριας συμβολοσειρών για την παρακάτω γραμματική:

$$\langle E \rangle ::= (\langle Y \rangle)$$
$$\langle Y \rangle ::= \langle A \rangle \langle B \rangle$$
$$\langle A \rangle ::= v | \langle E \rangle$$
$$\langle B \rangle ::= -\langle Y \rangle | +\langle Y \rangle | \varepsilon$$

Ταυτόχρονα το πρόγραμμα θα πρέπει να εκτυπώνει τα βήματα της κάθε παραγωγής.

Γραμματική

- Αλφάβητο: $V = \{ \langle E \rangle, \langle Y \rangle, \langle A \rangle, \langle B \rangle, (,), v, \varepsilon \}$
- $\langle E \rangle$: σύμβολο εκκίνησης ή αρχικό σύμβολο
 - Όλες οι προτάσεις πρέπει να προέρχονται από το σύμβολο εκκίνησης με διαδοχικές αντικαταστάσεις χρησιμοποιώντας τους κανόνες της γραμματικής (κανόνες παραγωγής).
- Τερματικά σύμβολα: $V_T = \{ (,), v, \varepsilon \}$
 - Εμφανίζονται σε προτάσεις της γραμματικής.
 - Δεν μπορούν να αναλυθούν περισσότερο.
- Μη-τερματικά σύμβολα: $V_N = \{ \langle E \rangle, \langle Y \rangle, \langle A \rangle, \langle B \rangle \}$
- $V = V_T \cup V_N$
- $V_T \cap V_N = \emptyset$

ΑΛΓΟΡΙΘΜΟΣ

Για τη παραγωγή μιας συμβολοσειράς από την παραπάνω γραμματική:

- Παίρνουμε ένα συντακτικό κανόνα (δηλαδή τον πρώτο), που το αριστερό του μέλος είναι το σύμβολο εκκίνησης (<E>).
- Στο δεξιό μέλος κάνουμε αντικατάσταση του αριστερού μέλους ενός οποιουδήποτε άλλου συντακτικού κανόνα που μπορούμε να βρούμε σ' αυτό το δεξιό μέλος με το αντίστοιχο δεξιό μέλος του κανόνα.
- Στο καινούριο δεξιό μέλος κάνουμε μια αντικατάσταση με τον ίδιο τρόπο.
- Συνεχίζουμε μέχρι να παραχθεί ένα δεξιό μέλος όπου όλα τα σύμβολα να είναι τερματικά.

Επισημάνσεις

- Για την τυχαία επιλογή των κανόνων αντικατάστασης μη τερματικών συμβόλων χρησιμοποιήθηκε η συνάρτηση `rand()` και `srand((unsigned int)time(NULL))` από τη βιβλιοθήκη (<random>) της c++. Η `srand()` δίνει "τυχαιοτητα" με βάση τη τρέχουσα ώρα ώστε να μη δίνονται οι ίδιοι αριθμοί κάθε φορά που τρέχουμε το πρόγραμμα.
- Αν εφαρμοστούν πάνω από 20 συντακτικοί κανόνες τότε κάνουμε αντικαταστάσεις των μη τερματικών συμβόλων που έχουν μείνει στη συμβολοσειρά με τους πιο "ευνοϊκούς" κανόνες (δηλαδή τα "<A>" με "ν" και τα "" με το κενό "ε") ώστε να λαμβάνεται υπόψιν ότι θα τερματίζεται η διαδικασία κάποια στιγμή όπως ζητείται από την εκφώνηση.

- Επειδή δεν μπορέσαμε να συμπεριλάβουμε ελληνικούς χαρακτήρες στην εκτύπωση των κανόνων παραγωγής που εφαρμόζονται αντικαταστήσαμε το "ε" με τον αγγλικό όρο "epsilon".

Παραδείγματα λειτουργίας:

```
C:\Users\xrist>cd C:\Users\xrist\Desktop\d
C:\Users\xrist\Desktop\d>g++ ASKSH2.cpp

C:\Users\xrist\Desktop\d>a
(<Y>) --> rule <E>::=<Y> was applied.
(<A><B>) --> rule <Y>::=<A><B> was applied.
(<E><B>) --> rule <A>::=<E> was applied.
(<E>+<Y>) --> rule <B>::=+<Y> was applied.
((<Y>)+<Y>) --> rule <E>::=<Y> was applied.
((<A><B>)+<Y>) --> rule <Y>::=<A><B> was applied.
((<A><B>)+<A><B>) --> rule <Y>::=<A><B> was applied.
((<E><B>)+<A><B>) --> rule <A>::=<E> was applied.
((<E><B>)+v<B>) --> rule <A>::=v was applied.
((<E>)+v<B>) --> rule <B>::=epsilon was applied.
((<E>)+v+<Y>) --> rule <B>::=+<Y> was applied.
(((<Y>)+v+<Y>) --> rule <E>::=<Y> was applied.
(((<A><B>)+v+<Y>) --> rule <Y>::=<A><B> was applied.
(((<A><B>)+v+<A><B>) --> rule <Y>::=<A><B> was applied.
(((v<B>)+v+<A><B>) --> rule <A>::=v was applied.
(((v<B>)+v+<E><B>) --> rule <A>::=<E> was applied.
(((v+<Y>)+v+<E><B>) --> rule <B>::=+<Y> was applied.
(((v+<Y>)+v+<E>+<Y>) --> rule <B>::=+<Y> was applied.
(((v+<Y>)+v+(<Y>)+<Y>) --> rule <E>::=<Y> was applied.
(((v+<A><B>)+v+(<Y>)+<Y>) --> rule <Y>::=<A><B> was applied.
(((v+<A><B>)+v+(<A><B>)+<Y>) --> rule <Y>::=<A><B> was applied.
(((v+<A><B>)+v+(<A><B>)+<A><B>) --> rule <Y>::=<A><B> was applied.
(((v+<A><B>)+v+(<A><B>)+<A><B>) --> rule <A>::=v was applied.
(((v+<A><B>)+v+(v<B>)+<A><B>) --> rule <A>::=v was applied.
(((v+<A><B>)+v+(v<B>)+v<B>) --> rule <A>::=v was applied.
(((v+v)+v+(v<B>)+v<B>) --> rule <B>::=epsilon was applied.
(((v+v)+v+(v+v<B>)) --> rule <B>::=epsilon was applied.
(((v+v)+v+(v+v)) --> rule <B>::=epsilon was applied.
```

```
C:\Users\xrist\Desktop\d>a
(<Y>) --> rule <E>::=<Y> was applied.
(<A><B>) --> rule <Y>::=<A><B> was applied.
(v<B>) --> rule <A>::=v was applied.
(v) --> rule <B>::=epsilon was applied.

C:\Users\xrist\Desktop\d>a
(<Y>) --> rule <E>::=<Y> was applied.
(<A><B>) --> rule <Y>::=<A><B> was applied.
(<E><B>) --> rule <A>::=<E> was applied.
(<E>) --> rule <B>::=epsilon was applied.
((<Y>)) --> rule <E>::=<Y> was applied.
((<A><B>)) --> rule <Y>::=<A><B> was applied.
((<E><B>)) --> rule <A>::=<E> was applied.
((<E>)) --> rule <B>::=epsilon was applied.
(((<Y>))) --> rule <E>::=<Y> was applied.
(((<A><B>))) --> rule <Y>::=<A><B> was applied.
(((<E><B>))) --> rule <A>::=<E> was applied.
(((<E>+<Y>))) --> rule <B>::=+<Y> was applied.
((((<Y>)+<Y>))) --> rule <E>::=<Y> was applied.
((((<A><B>)+<Y>))) --> rule <Y>::=<A><B> was applied.
((((<A><B>)+<A><B>))) --> rule <Y>::=<A><B> was applied.
((((v<B>)+<A><B>))) --> rule <A>::=v was applied.
((((v<B>)+<E><B>))) --> rule <A>::=<E> was applied.
((((v)+<E><B>))) --> rule <B>::=epsilon was applied.
((((v)+<E>+<Y>))) --> rule <B>::=+<Y> was applied.
((((v)+(<Y>)+<Y>))) --> rule <E>::=<Y> was applied.
((((v)+(<A><B>)+<Y>))) --> rule <Y>::=<A><B> was applied.
((((v)+(<A><B>)+<A><B>))) --> rule <Y>::=<A><B> was applied.
((((v)+(<A><B>)+<A><B>))) --> rule <A>::=v was applied.
((((v)+(<A><B>)+<A><B>))) --> rule <A>::=v was applied.
((((v)+(<A><B>)+<A><B>))) --> rule <B>::=epsilon was applied.
((((v)+(<A><B>)+<A><B>))) --> rule <B>::=epsilon was applied.
```

```

(<Y>) -->
(<A><B>) -->
(<E><B>) -->
(<E>+<Y>) -->
(((<Y>)+<Y>)) -->
(((<A><B>)+<Y>)) -->
(((<A><B>)+<A><B>)) -->
(((<E><B>)+<A><B>)) -->
(((<E><B>)+v<B>)) -->
(((<E>-<Y>)+v<B>)) -->
(((<E>-<Y>)+v+<Y>)) -->
((((<Y>)-<Y>)+v+<Y>)) -->
((((<A><B>)-<Y>)+v+<Y>)) -->
((((<A><B>)-<A><B>)+v+<Y>)) -->
((((<A><B>)-<A><B>)+v+<A><B>)) -->
(((v<B>)-<A><B>)+v+<A><B>)) -->
(((v<B>)-v<B>)+v+<A><B>)) -->
(((v<B>)-v<B>)+v+<E><B>)) -->
(((v-<Y>)-v<B>)+v+<E><B>)) -->
(((v-<Y>)-v+<Y>)+v+<E><B>)) -->
(((v-<Y>)-v+<Y>)+v+<E>)) -->
(((v-<Y>)-v+<Y>)+v+(<Y>)) -->
(((v-<A><B>)-v+<Y>)+v+(<Y>)) -->
(((v-<A><B>)-v+<A><B>)+v+(<Y>)) -->
(((v-<A><B>)-v+<A><B>)+v+(<A><B>)) -->
(((v-v<B>)-v+<A><B>)+v+(<A><B>)) -->
(((v-v<B>)-v+<A><B>)+v+(<A><B>)) -->
(((v-v<B>)-v+<A><B>)+v+(v<B>)) -->
(((v-v)-v+<A><B>)+v+(v<B>)) -->
(((v-v)-v+v)+v+(v<B>)) -->
(((v-v)-v+v)+v+(v))

```

```

rule <E>::=<(<Y>)> was applied.
rule <Y>::=<A><B> was applied.
rule <A>::=<E> was applied.
rule <B>::=<+<Y> was applied.
rule <E>::=<(<Y>)> was applied.
rule <Y>::=<A><B> was applied.
rule <Y>::=<A><B> was applied.
rule <A>::=<E> was applied.
rule <A>::=v was applied.
rule <B>::=<-<Y> was applied.
rule <B>::=<+<Y> was applied.
rule <E>::=<(<Y>)> was applied.
rule <Y>::=<A><B> was applied.
rule <Y>::=<A><B> was applied.
rule <Y>::=<A><B> was applied.
rule <A>::=v was applied.
rule <A>::=v was applied.
rule <A>::=<E> was applied.
rule <B>::=<-<Y> was applied.
rule <B>::=<+<Y> was applied.
rule <B>::=epsilon was applied.
rule <E>::=<(<Y>)> was applied.
rule <Y>::=<A><B> was applied.
rule <Y>::=<A><B> was applied.
rule <Y>::=<A><B> was applied.
rule <A>::=v was applied.
rule <A>::=v was applied.
rule <A>::=v was applied.
rule <B>::=epsilon was applied.
rule <B>::=epsilon was applied.
rule <B>::=epsilon was applied.

```

Οδηγίες χρήσης

Υποθέτοντας ότι έχουμε εγκαταστημένο έναν GCC compiler:

- ✓ Ανοίγουμε ένα παράθυρο cmd ή terminal όπου βρίσκεται στο path όπου είναι το αρχείο ASKSH2.cpp .
- ✓ Εκτελούμε την εντολή g++ -o CompilersTask ASKSH2.cpp.
- ✓ Το εκτελούμε με την εντολή ./CompilersTask σε terminal ή CompilersTask σε cmd.

Το πρόγραμμα είναι γραμμένο σε c++