

## Σχεδιασμός ενεργειών

# Προβλήματα σχεδιασμού ενεργειών

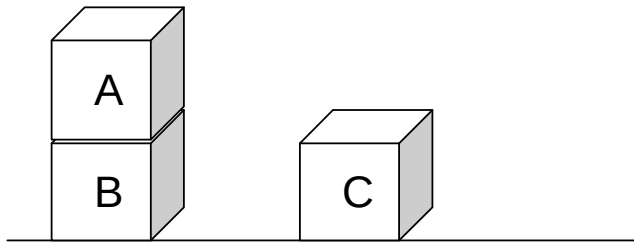
- Στα προβλήματα σχεδιασμού ενεργειών δίδονται
  - Η αρχική κατάσταση του κόσμου
  - Η επιθυμητή τελική κατάσταση του κόσμου
  - Το ρεπερτόριο των επιτρεπτών ενεργειών
- Ζητούμενο είναι η **ακολουθία ενεργειών** που, υποθέτοντας ότι θα εκτελεστούν επιτυχώς, θα μετατρέψουν τον κόσμο από την αρχική στην επιθυμητή τελική κατάσταση.
- Αυτή η ακολουθία ενεργειών ονομάζεται **σχέδιο**.

## Σχεδιασμός στο STRIPS (Fikes & Nilsson 1971)

- **Αδιαίρετες (και στιγμιαίες) ενέργειες:** οι ενέργειες δεν έχουν διάρκεια, θεωρείται ότι εκτελούνται στιγμιαία, δεν είναι δυνατή η διακοπή τους και η κατάσταση του κόσμου μπορεί να διαπιστωθεί μόνο πριν ή μετά την εκτέλεσή τους.
- **Ντετερμινιστικά αποτελέσματα:** τα αποτελέσματα κάθε ενέργειας είναι γνωστά εκ των προτέρων και εγγυημένα.
- **Υπόθεση κλειστού κόσμου:** δεν προστίθενται και δεν διαγράφονται οντότητες από τον κόσμο κατά τη διάρκεια του σχεδιασμού.
- **Στατικός κόσμος:** ο κόσμος μεταβάλλεται μόνο από τις ενέργειες του συστήματος σχεδισμού.

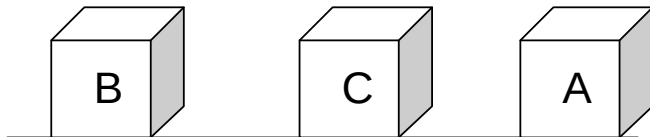
# Αναπαράσταση καταστάσεων στο STRIPS

τρέχουσα κατάσταση περιβάλλοντος



- Αρχική κατάσταση:  
onTable(c)  
onTable(b)  
on(a,b)  
clear(a)  
clear(c)

επιθυμητή κατάσταση περιβάλλοντος

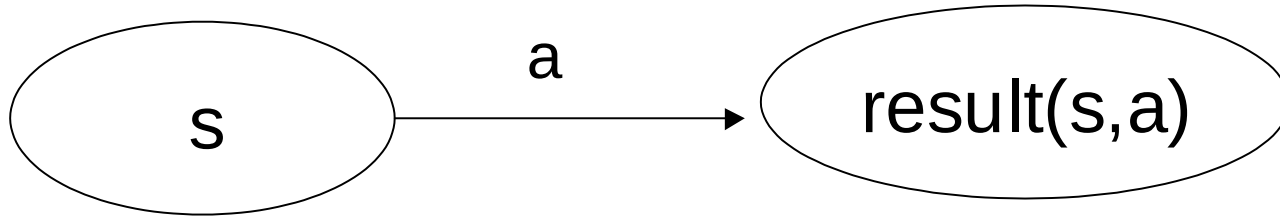


- Τελική κατάσταση:  
onTable(a)  
onTable(b)  
onTable(c)  
clear(a)  
clear(b)  
clear(c)

## Αναπαράσταση καταστάσεων

- Κάθε κατάσταση μπορεί να περιγραφεί
  - Με το σύνολο των ατομικών προτάσεων που αληθεύουν σε αυτή
  - Με τη σύζευξη των ατομικών προτάσεων που αληθεύουν σε αυτή
- Όσες προτάσεις δεν αναφέρονται ρητά σε μια κατάσταση μπορεί να θεωρηθούν ψευδείς (στατικός κόσμος, υπόθεση κλειστού κόσμου).

# Αναπαράσταση ενεργειών στο STRIPS



- Οι ενέργειες **μετασχηματίζουν** καταστάσεις.
- Κάθε ενέργεια προσδιορίζεται
  - Από τό (μοναδικό) όνομά της
  - Από το σύνολο **προϋποθέσεων** της: το σύνολο των προτάσεων που πρέπει να είναι αληθείς σε μια κατάσταση για να μπορεί να εκτελεστεί η ενέργεια σε αυτή την κατάσταση, **Pre(a)**.
  - Από το **σύνολο των αποτελεσμάτων** της:
    - το σύνολο των προτάσεων που **θα γίνουν αληθείς** στην κατάσταση που προκύπτει μετά την εκτέλεση της ενέργειας, **Add(a)**
    - το σύνολο των προτάσεων που θα πάψουν να είναι αληθείς στην κατάσταση που προκύπτει μετά την εκτέλεση της ενέργειας, **Del(a)**
- Περιορισμοί:
  - $\text{Pre}(a) \subseteq S$
  - $\text{Del}(a) \subseteq \text{Pre}(a)$
  - $\text{result}(s,a) = S - \text{Del}(a) \cup \text{Add}(a)$

## Παράδειγμα ορισμού ενεργειών

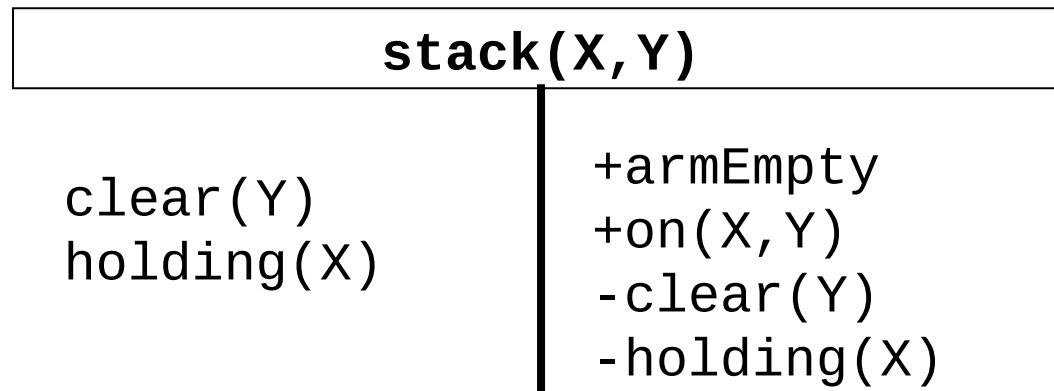
- `on(X, Y)`: object X is on object Y
- `clear(X)`: object X is clear
- `onTable(X)`: object X is on the table
- `holding(X)`: robot is holding object X
- `armEmpty`: robot is not holding anything

**stack(X, Y)**: robot places object X, which it is holding, on object Y

Pre {`clear(Y)`, `holding(X)`}

Del {`clear(Y)`, `holding(X)`}

Add {`armEmpty`, `on(X, Y)`}



# Άσκηση

- Να ορίσετε τις ενέργειες
- `unstack(X, Y) :` robot picks up object X from the top of object Y
- `pickup(X) :` robot picks up object X from the table
- `putdown(X) :` robot places object X, which it is holding, on the table



## Ορθότητα σχεδίου

- Ένα σχέδιο ενεργειών είναι μια ακολουθία ενεργειών  $\langle \alpha_1, \alpha_2, \dots, \alpha_n \rangle$
- Ένα σχέδιο είναι **ορθό** αν και μόνο αν
  - Η πρώτη του ενέργεια είναι εφαρμόσιμη στην αρχική κατάσταση του προβλήματος.
  - Κάθε ενέργεια του αφήνει τον κόσμο σε τέτοια κατάσταση ώστε να είναι εφαρμόσιμη η επόμενη ενέργεια του σχεδίου.
  - Η τελευταία ενέργειά του αφήνει τον κόσμο στην επιθυμητή τελική κατάσταση.

## Παράδειγμα ελέγχου ορθότητας

- Έστω το σχέδιο:

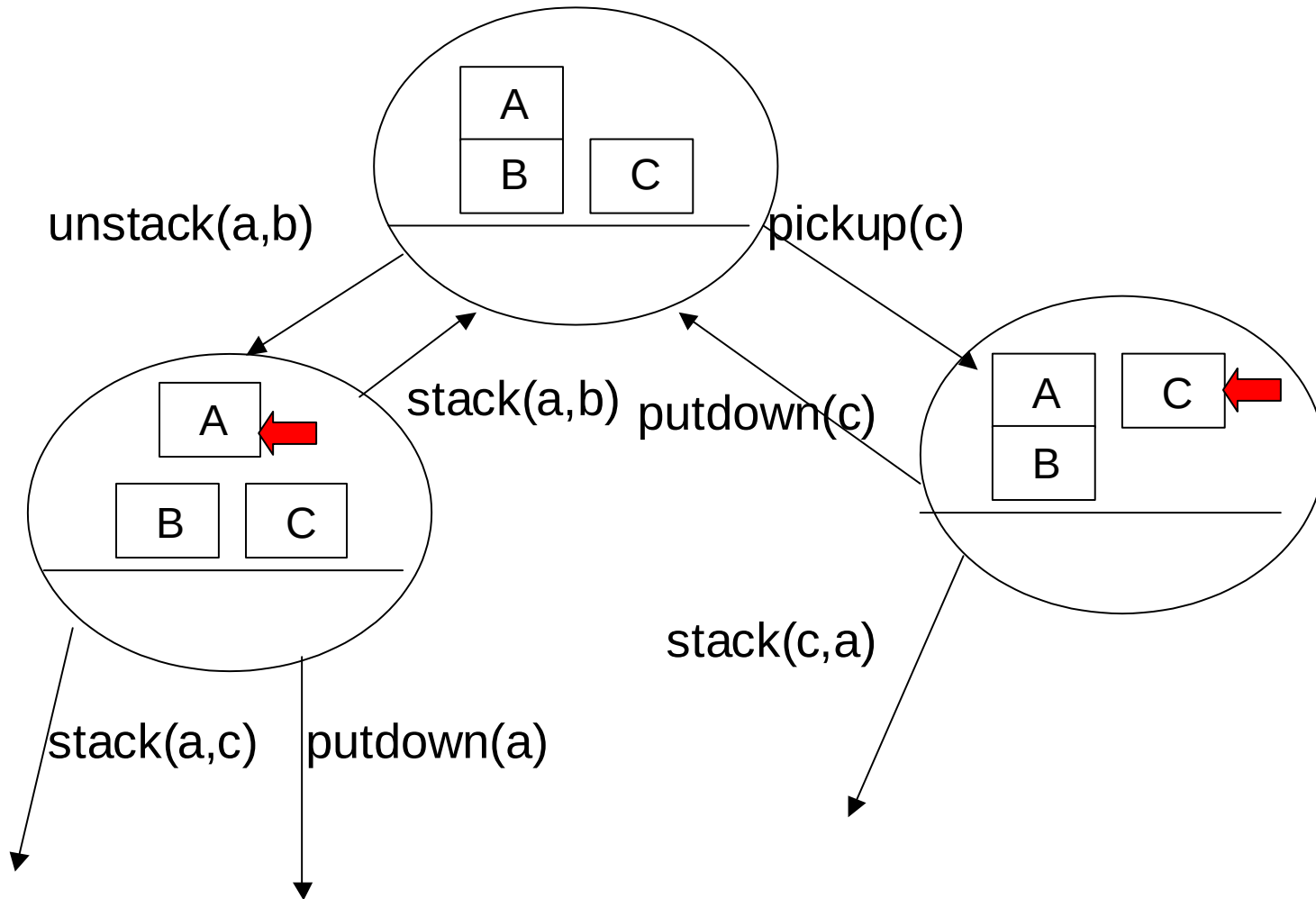
`<stack(c,a), unstack(c,a), pickup(b), putdown(b),  
putdown(c)>`

- Να ελέγξετε την ορθότητά του.

# Κατασκευή σχεδίου με αναζήτηση στο χώρο καταστάσεων

- Για την κατασκευή σχεδίου μπορεί να χρησιμοποιηθεί οποιοσδήποτε από τους αλγορίθμους αναζήτησης (τυφλή, ευριστική).
- Η κατασκευή του σχεδίου μπορεί να γίνεται
  - Προοδευτικά, από την αρχική προς την τελική κατάσταση: ορθή διάσχιση.
  - Ανάστροφα, από την τελική προς την αρχική κατάσταση: ανάστροφη διάσχιση.

## Σχεδιασμός με ορθή διάσχιση



## Σχεδιασμός με ανάστροφη διάσχιση

- Έστω  $G$  το σύνολο στόχων που πρέπει να επιτευχθούν (το σύνολο των προτάσεων που πρέπει να είναι αληθείς στην τελική κατάσταση).
- Επιλέγεται μια ενέργεια  $\alpha$  τέτοια ώστε
$$\mathbf{Del}(\alpha) \cap G \neq \emptyset \text{ και } \mathbf{Add}(\alpha) \cap G \neq \emptyset$$
- Η πιο αυστηρά, αν το  $G$  περιέχει πλήρη περιγραφή κατάστασης
$$\mathbf{Del}(\alpha) \cap G \neq \emptyset \text{ και } \mathbf{Add}(\alpha) \subseteq G$$
- Μετά την εφαρμογή αυτής της ενέργειας το  $G$  αναθεωρείται και γίνεται

$$G' = \mathbf{Pre}(\alpha) \cup G - \mathbf{Add}(\alpha)$$

- Η διαδικασία επαναλαμβάνεται για το νέο  $G'$ .
- Ένα σύνολο στόχων είναι **αδιέξοδο** αν
  - Περιέχει τουλάχιστον ένα γεγονός για το οποίο δεν υπάρχει ενέργεια που να το προσθέτει ή
  - Περιέχει σύνολο γεγονότων που είναι λογικά ασύμβατα
- Όταν διαπιστωθεί ότι ένα σύνολο στόχων είναι αδιέξοδο η διαδικασία σχεδιασμού **οπισθοδρομεί** στο πιο πρόσφατο σημείο επιλογής ενέργειας, για το οποίο υπάρχει εναλλακτική επιλογή.

# Παράδειγμα σχεδιασμού με ανάστροφη διάσχιση

- (στον πίνακα)