

PLANNING

Planning



Outline

01 Planning

02 Dunia Balok

Oxford Advanced Learners



Ide atau Metode



Diagram atau Peta



Penyusunan Benda



Penyusunan Keuangan

Planning



“

Apa itu
Planning

Planning



Planning



Kemampuan Dunia
Balok



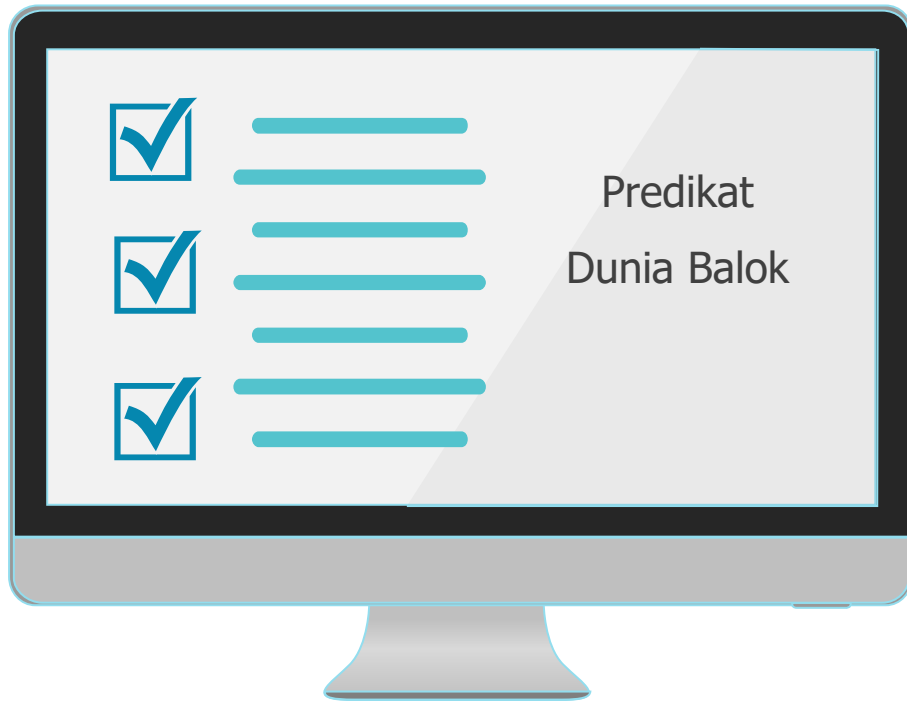
Daftar PAD

Planning



Dunia Balok (Blocs-
Worlds

Planning



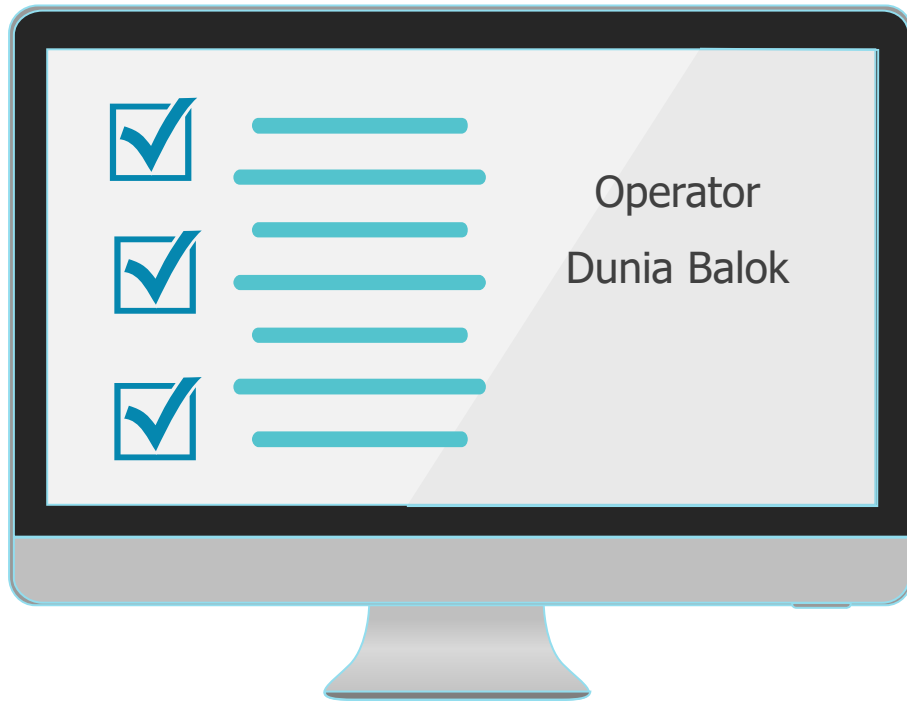
PREDIKAT UNTUK BALOK KOTAK

- ON (A,B)
- ON TABLE (A)
- CLEAR (A)

PREDIKAT UNTUK LENGAN ROBOT

- HOLDING (A)
- ARMEMPTY

Planning



Operator lengan robot

- STACK (A,B)
- UNSTACK (A,B)
- PICKUP (A)
- PUTDOWN (A)

GOAL STACK PLANNING

Goal State Planning (GSP)

Jadi

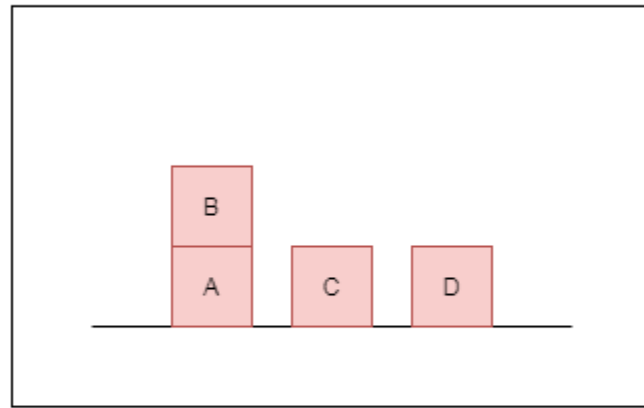
apasih

GSP PLANNING ?

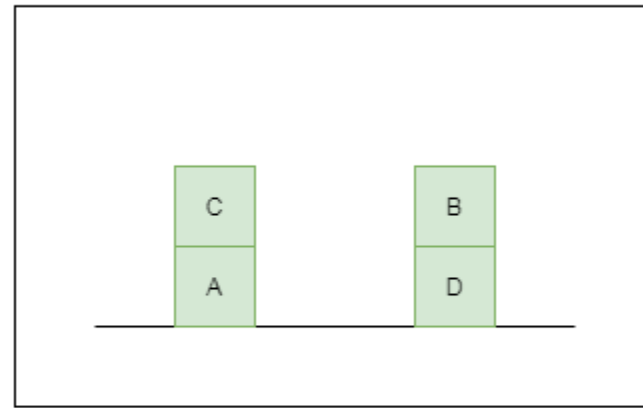
Goal State Planning (GSP)



Goal State Planning (GSP)



Initial State

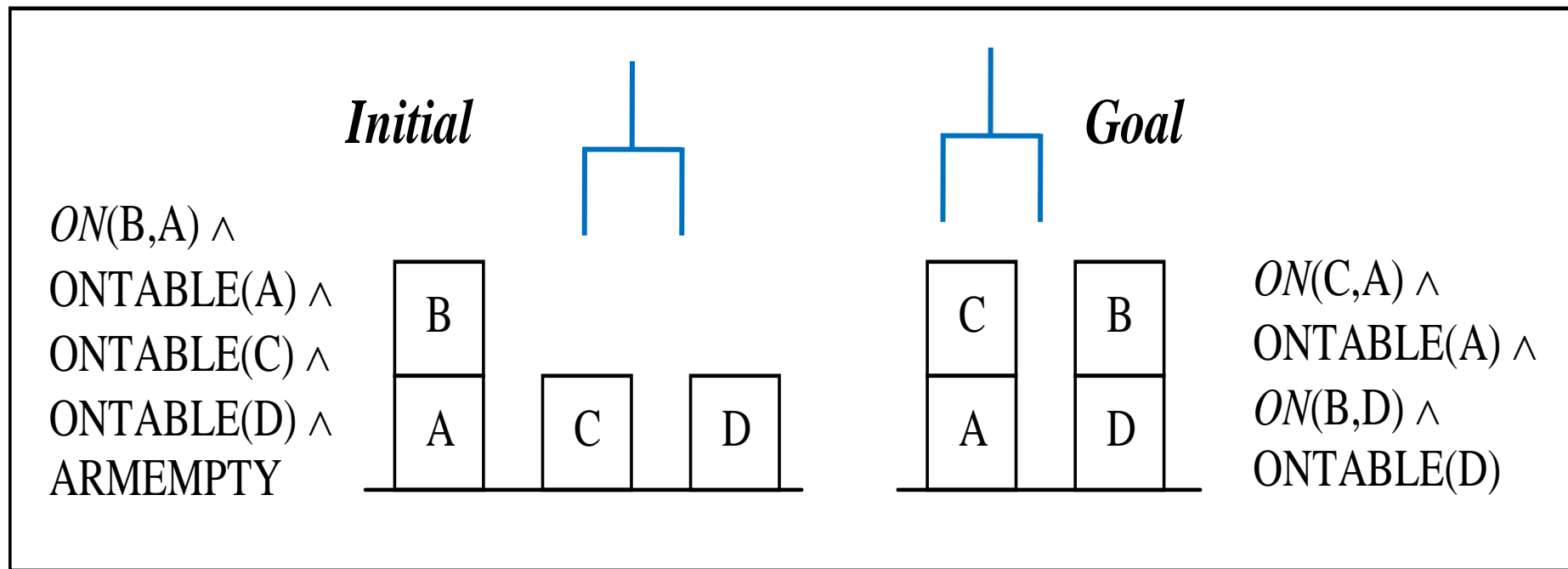


Goal State

Algoritma GSP

1. Tempatkan seluruh kondisi *goal-state* pada *slot stack* paling bawah.
 2. Masukkan setiap kondisi *goal-state* yang belum tercapai ke dalam sebuah *slot stack*.
 3. **Loop**
 - Keluarkan kondisi yang sudah dicapai dari dalam *stack*.
 - Ganti kondisi yang belum dicapai dengan operator yang sesuai.
 - Pindahkan operator yang bisa diaplikasikan ke dalam rencana penyelesaian.
 - Cek apakah *current-state* sama dengan *goal-state*
 - If** *current-state* = *goal-state* **Then**
 - sukses
 - End**
- End**

Contoh masalah



Contoh masalah

- **Initial State:**

$\text{ONTABLE}(A) \wedge \text{ONTABLE}(C) \wedge \text{ONTABLE}(D) \wedge \text{ON}(B,A) \wedge$
 ARMEMPTY

- **Goal State:**

$\text{ONTABLE}(A) \wedge \text{ONTABLE}(D) \wedge \text{ON}(B,D) \wedge \text{ON}(C,A) \wedge \text{ARMEMPTY}$

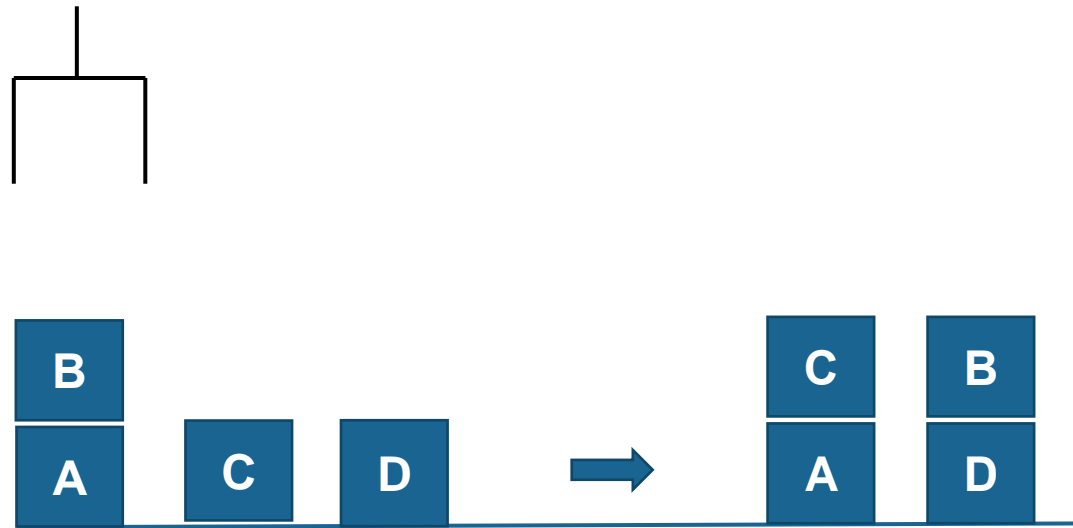
- **Current State = Initial State**

- **Stack paling bawah = Goal State**

Penyelesaian masalah

Isi stack Langkah 1

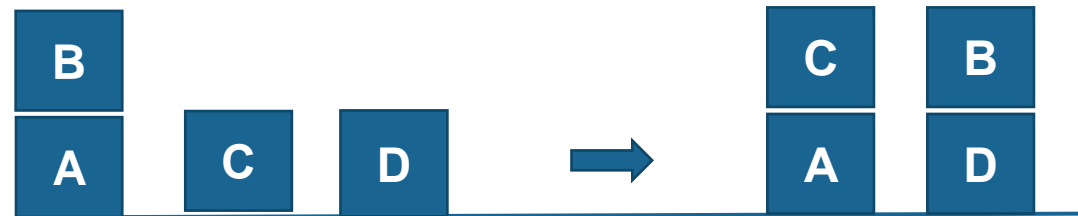
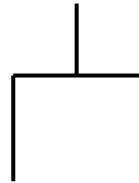
ON (C,A)
ON (B,D)
ONTABLE (A)
ONTABLE (D)



Penyelesaian masalah

Isi stack Langkah 2

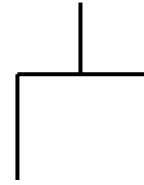
CLEAR (A)
HOLDING (C)
CLEAR (A)^HOLDING (C)
STACK (C,A)
ON (B,D)
ON (C,A)^ON (B,D)^OTAD



Penyelesaian masalah

Isi stack Langkah 3

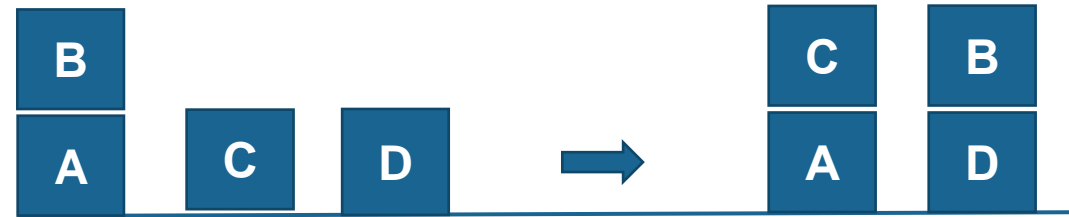
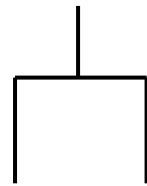
ON (B,A)
CLEAR (B)
ARMEMPTY
ON (B,A)^CLEAR (B)^ARMEMPTY
UNSTACK (B,A)
HOLDING (C)
STACK (C,A)
ON (B,D)



Penyelesaian masalah

Current state Langkah 4

ONTABLE (A)
ONTABLE (C)
ONTABLE (D)
HOLDING (B)



Penyelesaian masalah

Isi stack Langkah 5

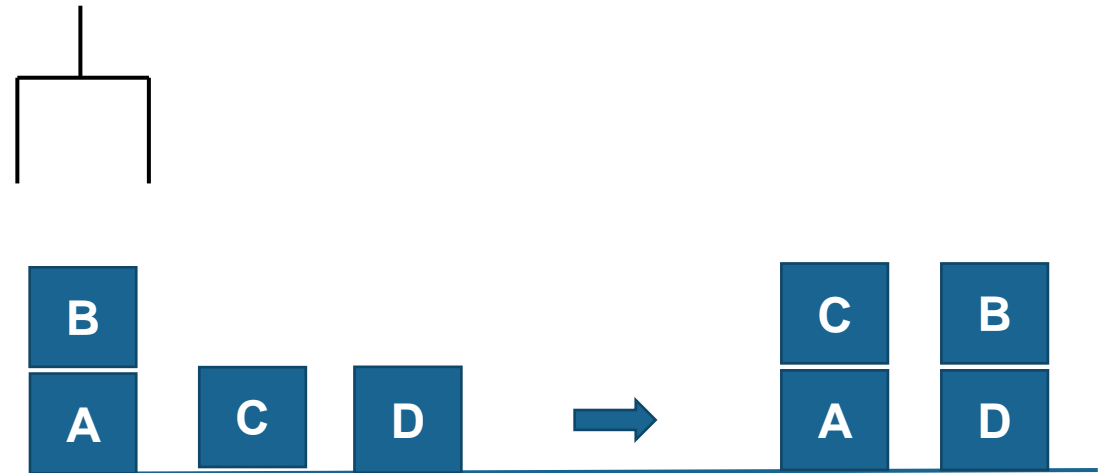
HOLDING (C)

CLEAR (A)^HOLDING (C)

STACK (C,A)

ON (B,D)

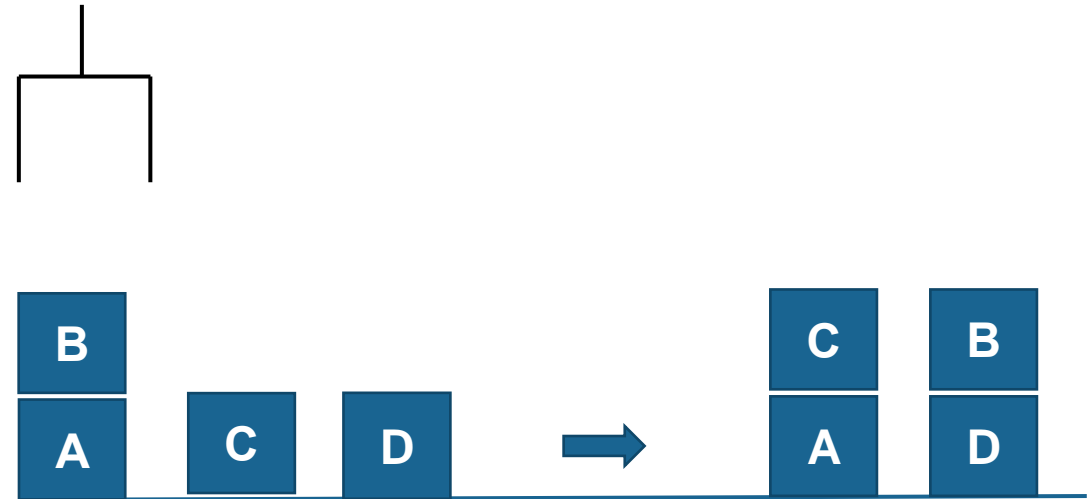
ON (C,A)^ON (B,D)^OTAD



Penyelesaian masalah

Isi stack Langkah 6

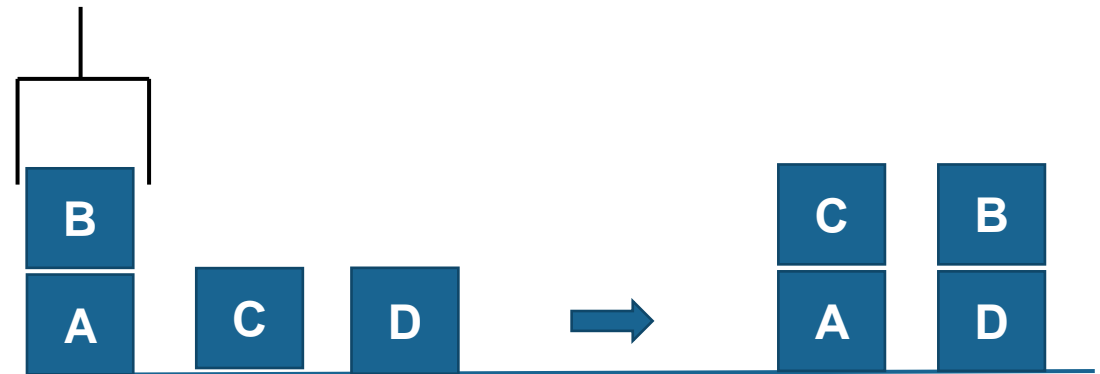
ONTABLE (C)
CLEAR (C)
ARMEMPTY
ONTABLE (C)^CLEAR (C)^ARMEMPTY
PICKUP (C)
CLEAR (A)^HOLDING (C)
STACK (C,A)
ON (B,D)
ON (C,A)^ON (B,D)^OTAD



Penyelesaian masalah

Isi stack Langkah 7

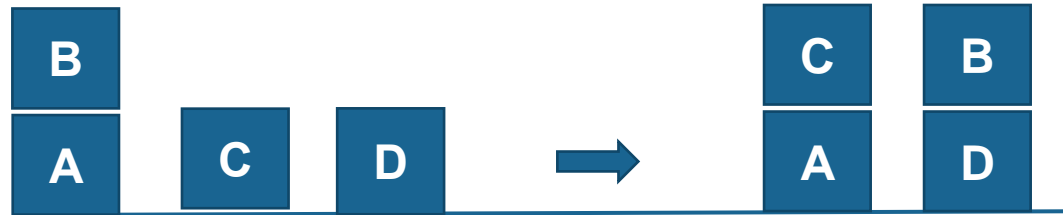
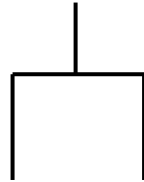
CLEAR (D)
HOLDING (B)
CLEAR (D)^HOLDING (B)
STACK (B,D)
ONTABLE (C)^CLEAR (C)^ARMEMPTY
PICKUP (C)
CLEAR (A)^HOLDING (C)
STACK (C,A)
ON (B,D)
ON (C,A)^ON (B,D)^OTAD



Penyelesaian masalah

Rencana penyelesaian

UNSTACK (B,A)
STACK (B,D)
PICKUP (C)
STACK (C,A)



CONSTRAINT POSTING

Constraint Posting



Apa itu
Constraint Posting?

Constraint Posting

Fungsi – Fungsi Pemandu Constraint Posting :



Step-addition



Simple -
establishment



Promotion



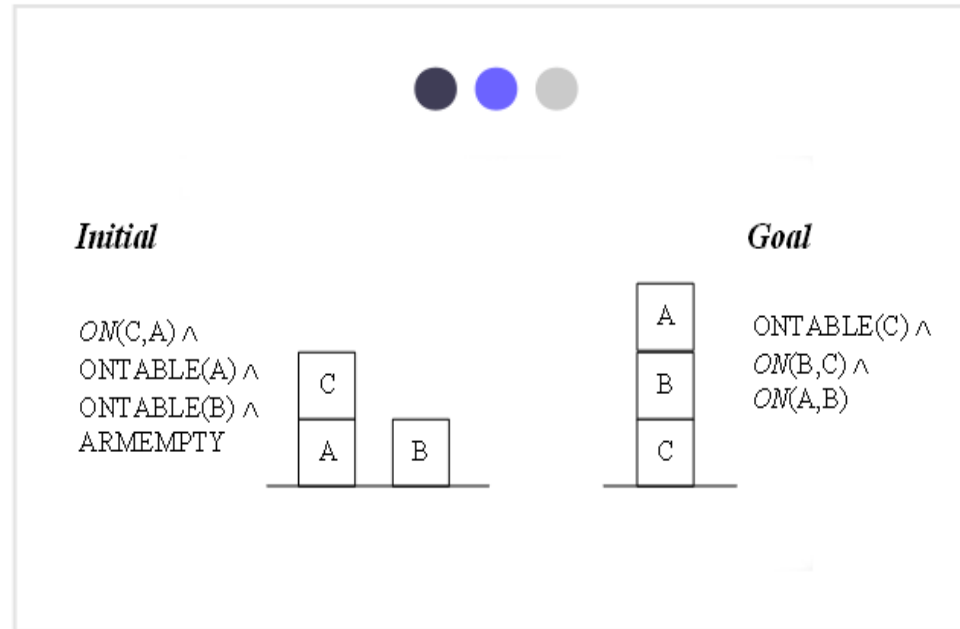
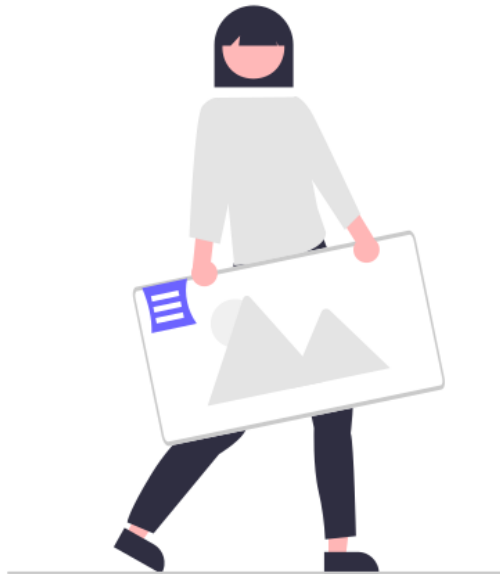
Separation



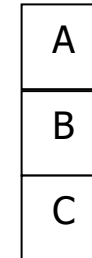
Declobbering



Contoh Kasus



ON (A,B)



STACK (A,B)

ON (B,C)

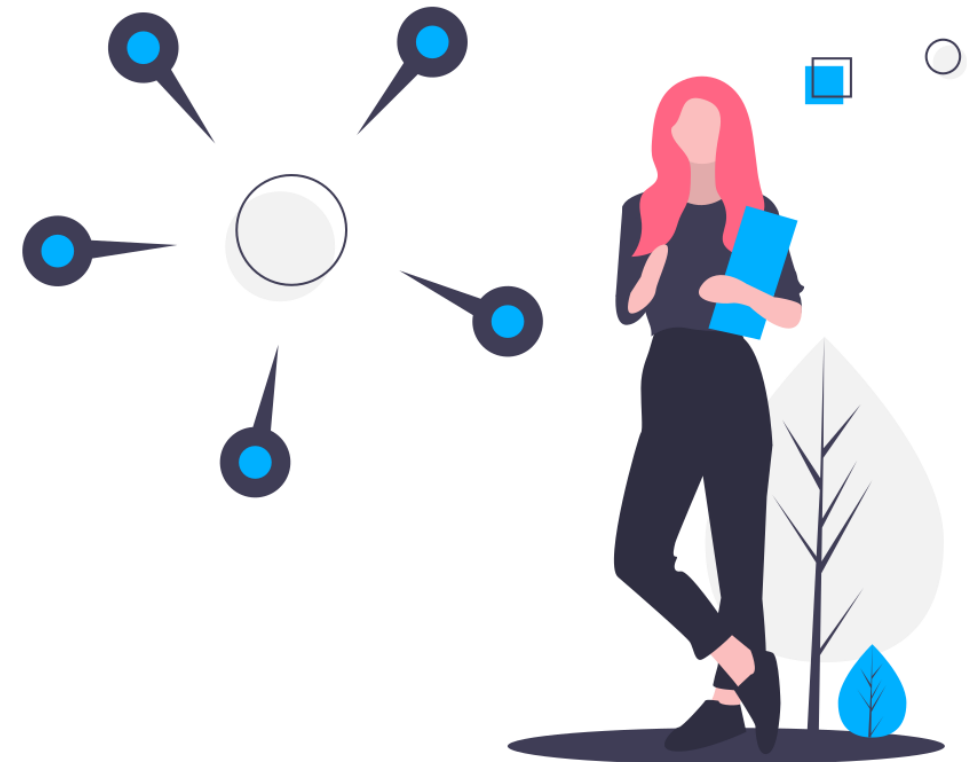


STACK(B,C)

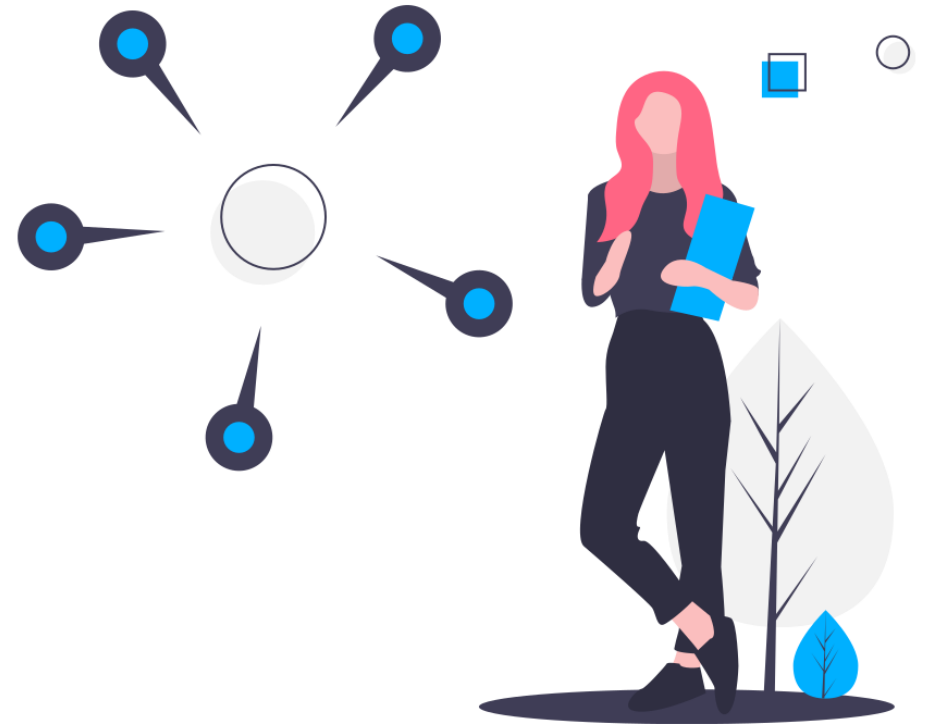
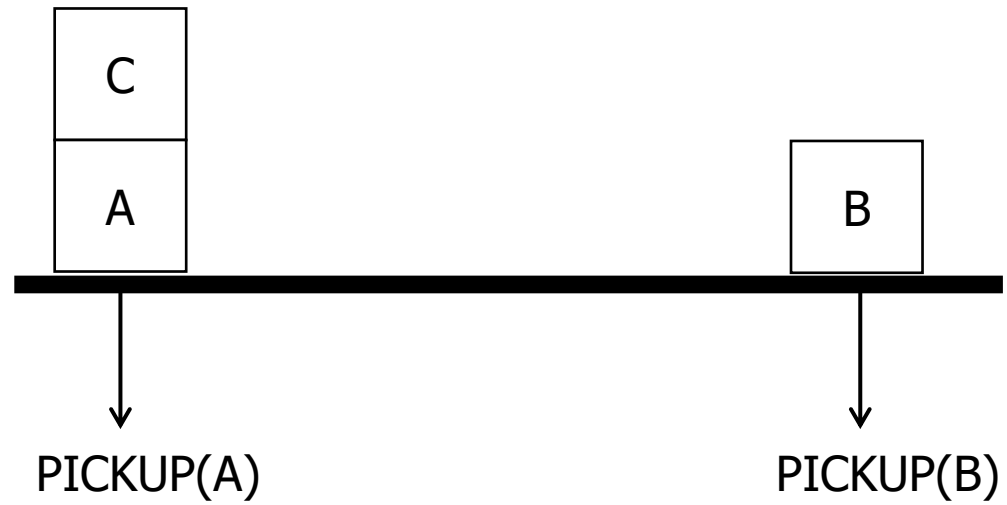
Contoh Kasus

| | |
|---|---|
| $CLEAR(B)$ $*HOLDING(A)$ | $CLEAR(C)$ $*HOLDING(B)$ |
| $STACK(A,B)$ | $STACK(B,C)$ |
| $ARMEMPTY$ $ON(A,B)$ $\neg CLEAR(B)$ $\neg HOLDING(A)$ | $ARMEMPTY$ $ON(B,C)$ $\neg CLEAR(C)$ $\neg HOLDING(B)$ |

- * Menunjukkan kondisi yang belum tercapai agar operator bisa di operasikan.
- ▮ Menunjukkan kondisi yang dihilangkan setelah operator di aplikasikan.



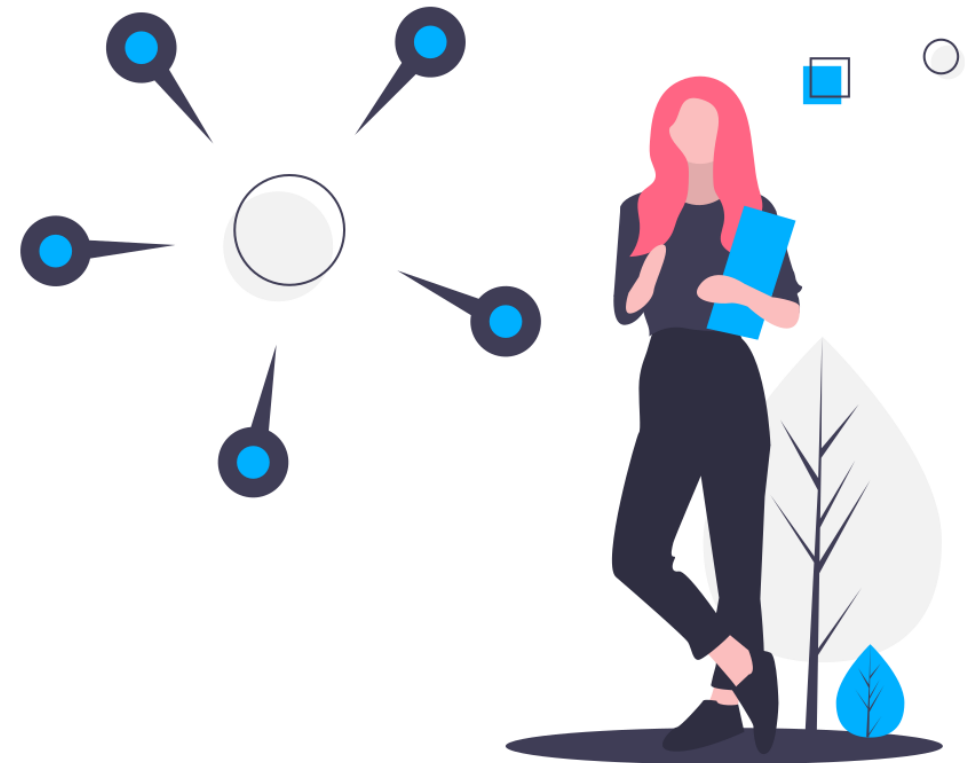
Contoh Kasus



Contoh Kasus

| | |
|---|---|
| *CLEAR(A) ONTABLE(A) *ARMEMPTY | *CLEAR(B) ONTABLE(B) *ARMEMPTY |
| <hr/> | <hr/> |
| PICKUP(A) | PICKUP(B) |
| <hr/> | <hr/> |
| HOLDING(A) $\neg \text{ONTABLE(A)}$ $\neg \text{ARMEMPTY}$ | HOLDING(B) $\neg \text{ONTABLE(B)}$ $\neg \text{ARMEMPTY}$ |

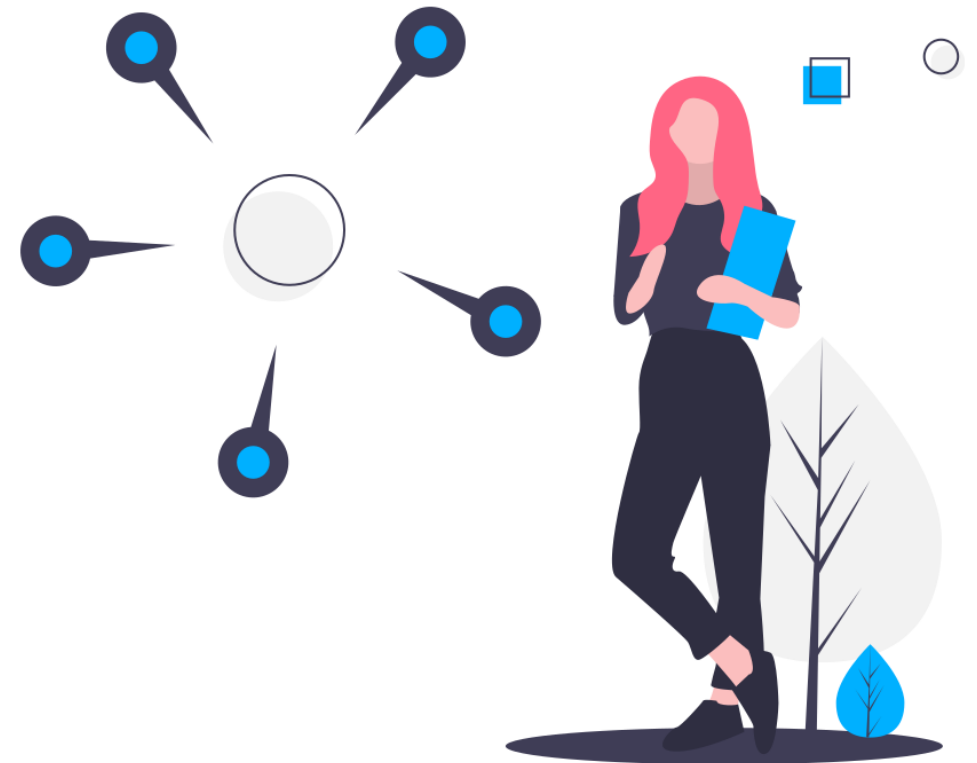
└─ ARMEMPTY CLEAR(A) CLEAR(B)



Contoh Kasus

```
PICKUP(A) ← STACK(A,B)  
PICKUP(B) ← STACK(B,C)  
PICKUP(B) ← STACK(A,B)
```

*CLEAR(B), CLEAR(B)



Contoh Kasus

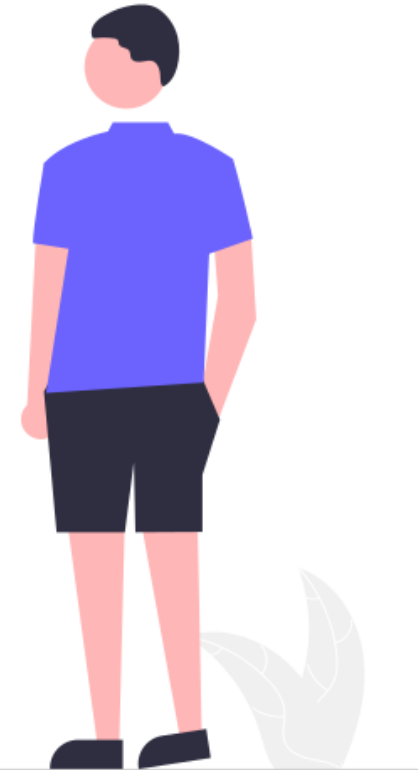


*ARMEMPTY : precondition PICKUP(A) dan PICKUP(B).

*CLEAR(A) : precondition PICKUP(A).

Mencapai *ARMEMPTY pada langkah PICKUP(A) dan PICKUP(B).

PICKUP(A) \leftarrow PICKUP(B).

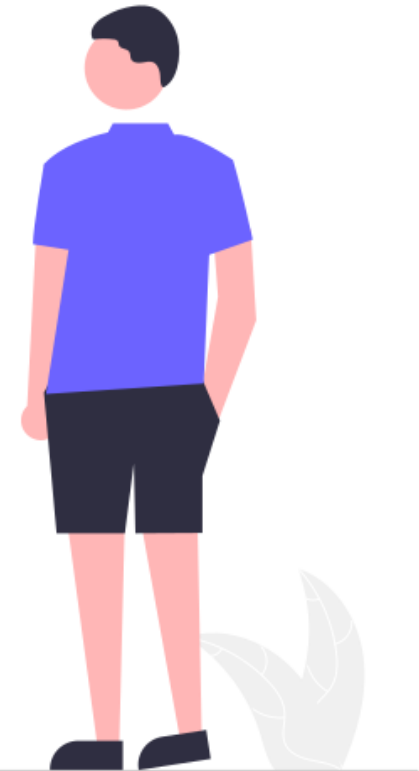


Contoh Kasus

PICKUP(A) \leftarrow PICKUP(B).

~~STACK(A,B)~~

~~PICKUP(A) \rightarrow PUTDOWN(A) \leftarrow PICKUP(B).~~



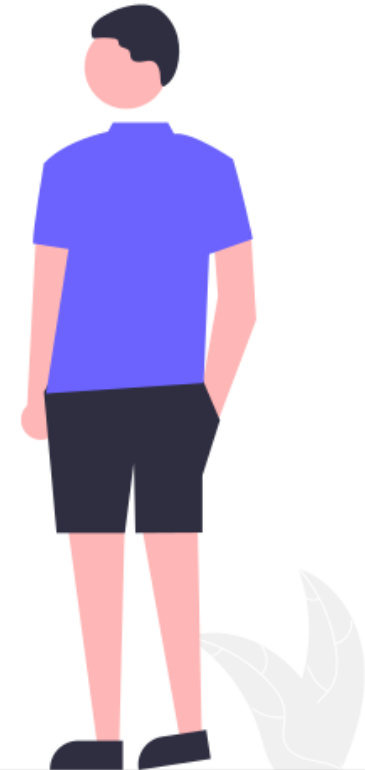
Contoh Kasus



PICKUP(B) \leftarrow PICKUP(A).

PICKUP(B) \leftarrow STACK(B,C) \leftarrow PICKUP(A).

*CLEAR(A)



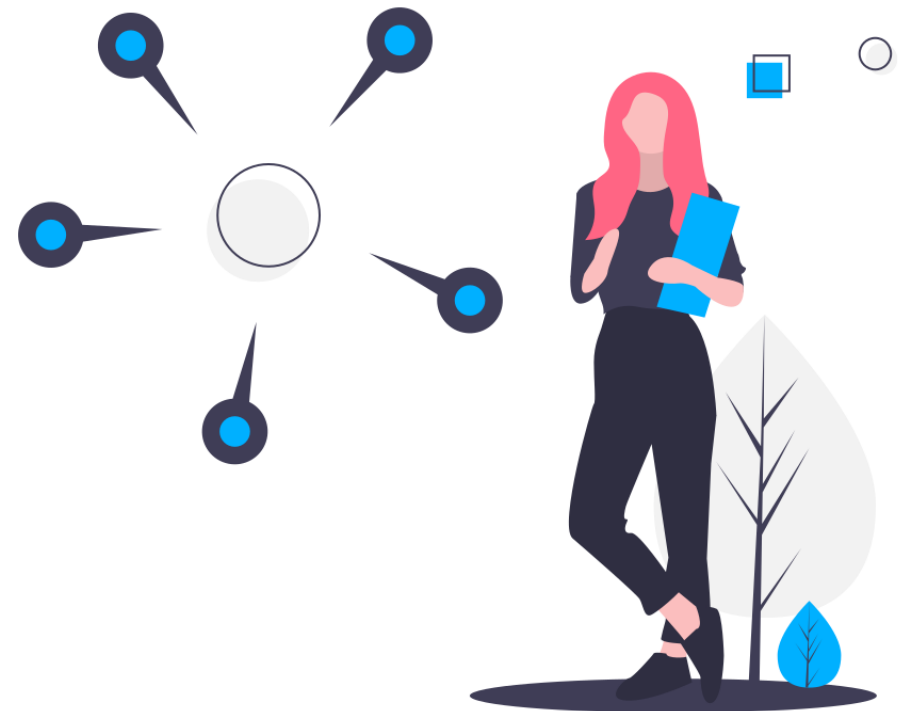
Contoh Kasus

| |
|---|
| *CLEAR(x) *ON(x,A) *ARMEMPTY |
| UNSTACK(x,A) |
| HOLDING(C) CLEAR(A) \neg ON(x,A) \neg ARMEMPTY |


UNSTACK(X,A) \leftarrow STACK(B,C).

UNSTACK(X,A) \leftarrow STACK(A).

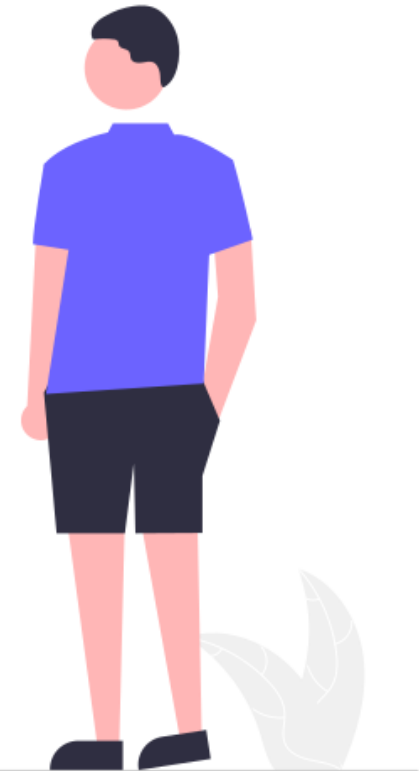
UNSTACK(X,A) \leftarrow STACK(B).



Contoh Kasus



PICKUP(B) \Leftrightarrow ARMEMPTY,
+ UNSTACK(C,A) \rightarrow ARMEMPTY,
ARMEMPTY \rightarrow PICKUP(A)



Contoh Kasus

HOLDING(C)

PUTDOWN(C)

ONTABLE(C)
ARMEMPTY
 \neg HOLDING(C)

UNSTACK(C,A)

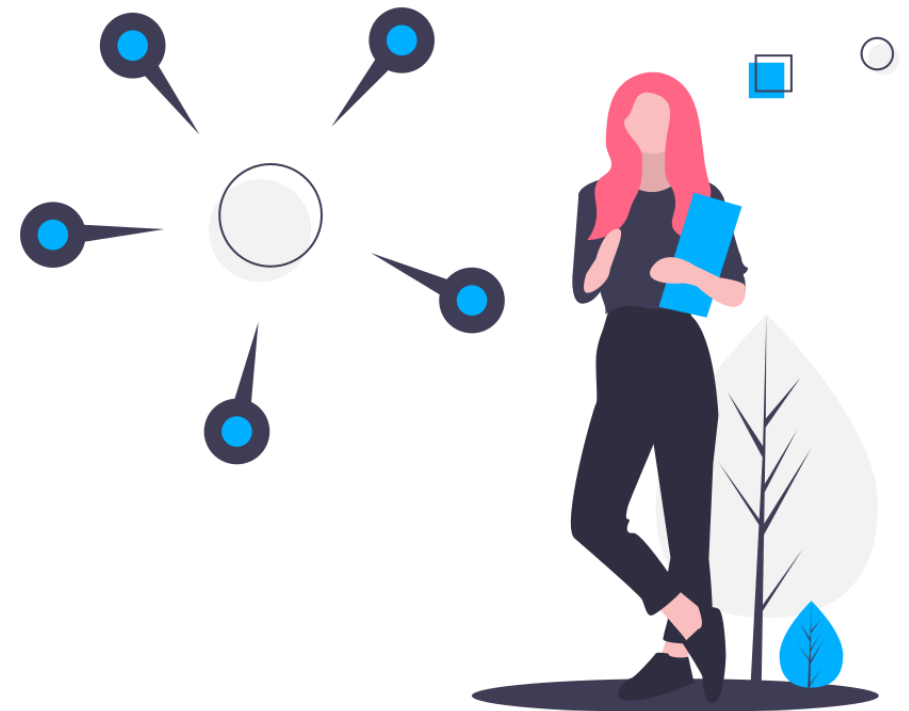
PUTDOWN(C)

PICKUP(B)

STACK(B,C)

PICKUP(A)

STACK(A,B)



Constraint Posting

- ✓ Setiap penambahan langkah baru pada CP membutuhkan pengecekan.
- ✓ Setiap pengurutan langkah baru pada CP membutuhkan pembandingan.



Constraint Posting

- ✓ Algoritma ini membutuhkan fungsi – fungsi pemandu.
- ✓ Pendefinisian setiap variabel dan struktur data di dukung oleh fungsi – fungsi pemandu.



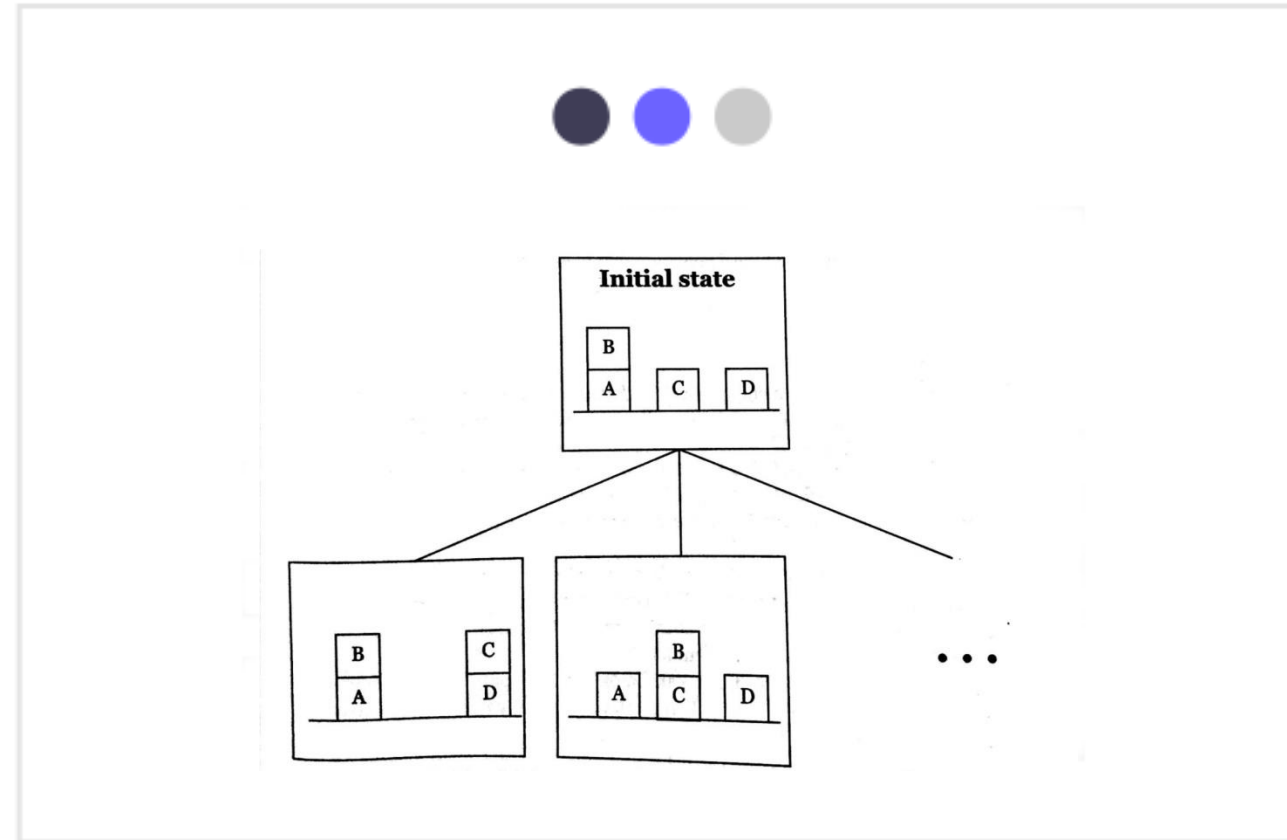
Constraint Posting



“ Algoritma CP

1. Inisialisasi S sebagai satu set proposisi (kondisi pada goal state).
2. Hilangkan beberapa proposisi yang belum tercapai P dari S
3. Capai P dengan menggunakan salah satu fungsi pemandu yang ada
4. Rview slurung langkah untuk melihat precondition yang belum tercapai
tambahan frecondition yang belum tercapai kedalam S
5. Jika S kosong, lengkapi rencana penyelesaian dengan mengkonversi urutan
sebagian langkah menjadi urutan lengkap
6. Jika tidak kosong, kembali ke langkah 2.

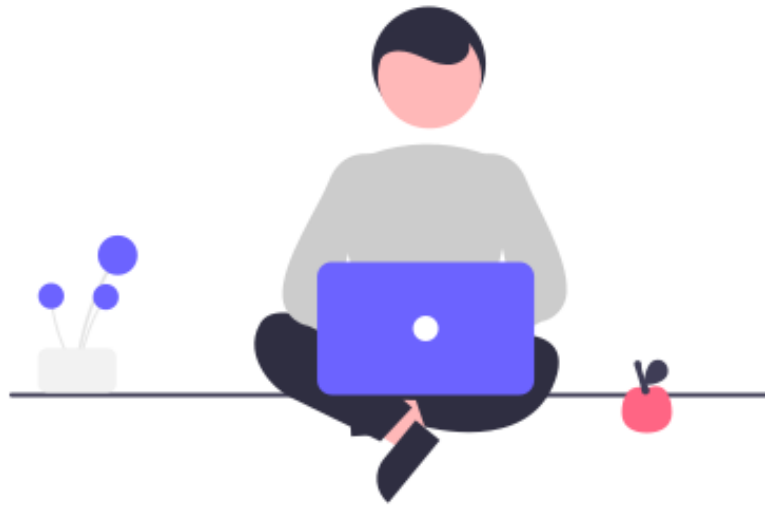
Algoritma CP [RIC91]



Constraint Posting

Kesimpulan

- Planning adalah suatu teknik penyelesaian masalah yang hanya bisa digunakan untuk menyelesaikan masalah yang dapat di komposisi.
- GSP adalah metode planning yang paling sederhana yang hanya menggunakan satu stack
- CP adalah metode planning secara umum bisa menemukan solusi yang lebih efisien dibanding solusi yang dihasilkan oleh gsp



Constraint Posting

Selesai