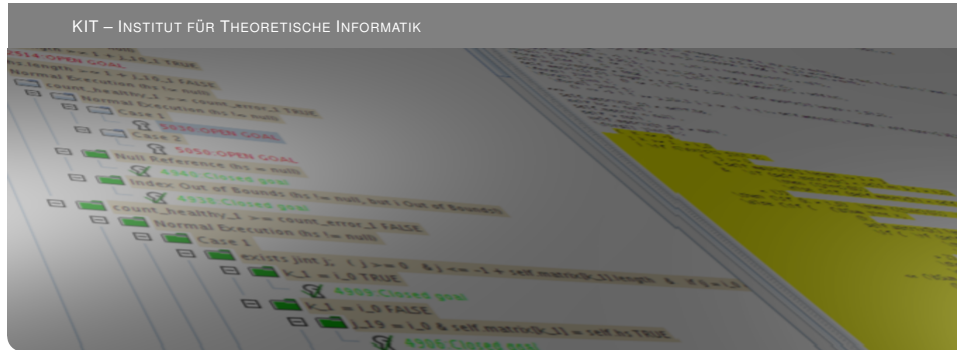


Formale Systeme

Prof. Dr. Bernhard Beckert, WS 2018/2019

Zweite Praxisaufgabe: Auftragsplanung mit SMT

KIT – INSTITUT FÜR THEORETISCHE INFORMATIK



Eingabe

Eine Menge $J = \{j_1, \dots, j_n\}$ von Aufträgen wobei jeder Auftrag

- ▶ eine Dauer,
- ▶ eine zugewiesene Maschine, und
- ▶ ein Fälligkeitstermin (deadline) besitzt.

Eine Vorgängerrelation $\prec: J \times J$:

$i \prec j$ gdw. j darf erst nach Ende von i anfangen

Ausgabe

Startdatum b_j für jeden Auftrag, sodass

- ▶ das Fälligkeitsdatum jedes Auftrages eingehalten wird,

Ausgabe

Startdatum b_j für jeden Auftrag, sodass

- ▶ das Fälligkeitsdatum jedes Auftrages eingehalten wird,
- ▶ zu jedem Zeitpunkt maximal nur ein Auftrag auf jeder Maschine läuft,

Ausgabe

Startdatum b_j für jeden Auftrag, sodass

- ▶ das Fälligkeitsdatum jedes Auftrages eingehalten wird,
- ▶ zu jedem Zeitpunkt maximal nur ein Auftrag auf jeder Maschine läuft,
- ▶ kein Auftrag gestartet wird, bevor nicht alle seine Vorgänger fertig gestellt wurden, und

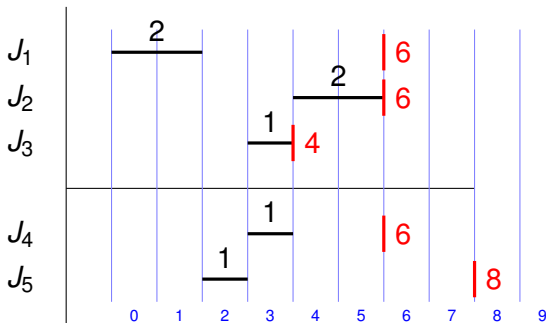
Ausgabe

Startdatum b_j für jeden Auftrag, sodass

- ▶ das Fälligkeitsdatum jedes Auftrages eingehalten wird,
- ▶ zu jedem Zeitpunkt maximal nur ein Auftrag auf jeder Maschine läuft,
- ▶ kein Auftrag gestartet wird, bevor nicht alle seine Vorgänger fertig gestellt wurden, und
- ▶ es keine Arbeitspausen gibt, d.h. zu jedem Zeitpunkt existiert ein Auftrag, der abgearbeitet wird (oder alle Aufträge sind erfüllt).

Beispiel

	Dauer	Maschine	Fälligkeit	Vorgänger
J_1	2	1	6	
J_2	2	1	6	
J_3	1	1	4	J_5
J_4	1	2	6	J_1
J_5	1	2	8	J_1



1. Formalisieren Sie das gegebene in SMT-LIB:
 - ▶ Signatur
 - ▶ Assertions

Abgabe: SMT-LIB Datei.

¹Java 11, Python 3.6

1. Formalisieren Sie das gegebene in SMT-LIB:

- ▶ Signatur
- ▶ Assertions

Abgabe: SMT-LIB Datei.

2. Schreiben Sie ein Programm¹, das eine Probleminstance einliest und SMT-LIB Eingabe erzeugt.

Abgabe: Quelltext

¹Java 11, Python 3.6

1. Formalisieren Sie das gegebene in SMT-LIB:

- ▶ Signatur
- ▶ Assertions

Abgabe: SMT-LIB Datei.

2. Schreiben Sie ein Programm¹, das eine Probleminstance einliest und SMT-LIB Eingabe erzeugt.

Abgabe: Quelltext

1. Hinweis: Beginnen Sie mit Ihrer Formalisierung

¹Java 11, Python 3.6

1. Formalisieren Sie das gegebene in SMT-LIB:

- ▶ Signatur
- ▶ Assertions

Abgabe: SMT-LIB Datei.

2. Schreiben Sie ein Programm¹, das eine Probleminstance einliest und SMT-LIB Eingabe erzeugt.

Abgabe: Quelltext

1. Hinweis: Beginnen Sie mit Ihrer Formalisierung

2. Hinweis: Versuchen Sie zu Optimieren

(Vorschläge: Quantoren ausrollen, Bitvektor, Enum datentyp)

¹Java 11, Python 3.6