

– Praktikumsaufgabe 7 –

Thema: *Ratenmonotones Scheduling*

Zielstellung: Vertiefung bzw. Wiederholung relevanter Aspekte des Ratenmonotonen Scheduling, Praxisbeispiel zur Prioritätsinversion

1. Ein System enthalte 4 periodische Tasks $T_i(t_p, t_e)$ mit $T_1(8, 1)$, $T_2(15, 3)$, $T_3(20, 4)$ und $T_4(22, 6)$. Konstruieren Sie den Schedule nach RMS im Intervall $[0, 50]$!
2. Gegeben sind jeweils 3 unabhängige periodische Tasks $T_i(t_p, t_e)$.
 - a) Zeigen Sie, dass $(10, 2)$, $(15, 5)$, $(25, 9)$ ratenmonoton einplanbar sind!
 - b) Zeigen Sie, dass $(10, 2)$, $(12, 5)$, $(15, 4)$ nicht ratenmonoton einplanbar sind!
3. Implementieren Sie ein Werkzeug zur Zeitanforderungsanalyse. Die Taskbeschreibung soll die Parameter $t_{\varphi, i}$, $t_{p, i}$, $t_{e, i}$, $t_{d, i}$ sowie den Index der Tasks aus einer Textdatei entnehmen und in etwa folgende Ausgabe generieren:

```
robge@hadrian:~/src/tda$ cat tasks3.txt
T1(0, 4, 1, 4)
T2(0, 5, 2, 7)
T3(0, 20, 5, 20)
robge@hadrian:~/src/tda$ ./tda < tasks3.txt
  Task      t_phi    t_p     t_e     t_d      u_i      t_mr    ok
T(001)    0.00     4.00    1.00    4.00    0.2500    1.00    v
T(002)    0.00     5.00    2.00    7.00    0.4000    3.00    v
T(003)    0.00    20.00    5.00   20.00    0.2500   15.00    v
u = 0.90000
```

Prüfen Sie dann,

- a) ob die gegebene Taskmenge `tasks6.txt` ratenmonoton planbar ist,
 - b) falls ja, ob eine weitere Task mit der Periode $t_p = 71$ hinzugenommen werden könnte, und (wiederum) falls ja, wie groß deren maximale Abarbeitungszeit betragen könnte.
- 4.* Ermitteln Sie für `tasks6.txt` und für `tasks-100-0.8.txt` (100 Tasks, $u = 0.8$), wieviel Zeiteinheiten für Nicht-Echtzeit-Tasks zur Verfügung stehen („Wieviel Luft ist noch im generierten Plan?“).
 5. Finden Sie heraus, was auf dem Mars wirklich geschah!
(<http://www.cs.cornell.edu/courses/cs614/1999sp/papers/pathfinder.html>)
Welches Phänomen führte zu dem beschriebenen Fehlverhalten?