

GKI - Hausaufgaben 1

Tao Xu, - Mitja Richter, 324680

Aufgabe 1

1.a)

Zustandsraum:

$S = (a_1, a_2, a_3, a_4, a_5, z), a_i \in \{0, 1, 10, 100\}$ und $z \in [0, 127]$

wobei a_i für die jeweils zu bearbeitende Aufgabe steht und 0, 1, 10, 100 für unbearbeitet(0), Alfons(1), Bernd(10) und Christine(100) (als bearbeitende Personen). Z stehen für die noch aufzuwendende Bearbeitungszeit von Alfons, Bernd und Christine. Es wird also ein 6-stelliges 7bit-Array benötigt.

Startzustand:

$S_0 = (0, 0, 0, 0, 0, 0)$

Zielzustand:

$S_Z = (a_1^Z, a_2^Z, a_3^Z, a_4^Z, a_5^Z, z^Z)$, mit $a_i^Z \neq 0$ und $z^Z = 0$

Aktionen: Die Auswahl des Studenten der die nächste noch nicht bearbeitete Aufgabe bearbeiten soll. Dargestellt mithilfe der Überföhrungsfunktion $wahl$ mit dem Parameter Student (s).

$wahl(s) : S = (a_1, a_2, a_3, a_4, a_5, z) \rightarrow S' = (a'_1, a'_2, a'_3, a'_4, a'_5, z'), s \in \{0, 1, 10, 100\}$

sodass gilt wenn $s \in \{1, 10, 100\}$ (d.h. wenn ein Student für eine Aufgabe ausgewählt wird):

sei a_j das erste $a_i \neq 0$ für mit $<$ geordneten i und $/$ die Ganzzahldivision

- $\forall a'_i, i \neq j. a'_i = a_i$
- $a'_j = s$ (Belegen der Aufgabe mit Alfons(1), Bernd(10) oder Christine(100))
- $z' = s + z$ (Bilden der neuen Bearbeitungszeit)
- $\frac{z'}{100} \leq 1$ (Christine darf nicht zwei Aufgaben gleichzeitig machen)
- $\frac{(z' \bmod 100)}{10} \leq 2$ (Bernd darf nicht zwei Aufgaben gleichzeitig machen)
- $((z' \bmod 100) \bmod 10) \leq 4$ (Alfons darf nicht zwei Aufgaben gleichzeitig machen)
- $a'_1 \neq 100$ (Christine kann Aufgabe 1 nicht)
- $a'_3 \neq 100 \wedge a'_3 \neq 10$ (Christine und Bernd können Aufgabe 3 nicht)
- $a'_4 \neq 100$ (Christine kann Aufgabe 4 nicht)

und wenn $s = 0$ (d.h. wenn kein neuer Student ausgewählt wird):

- $\forall a'_i. a'_i = a_i$
- $z' = z - \text{sgn}(\frac{z}{100}) \cdot 100 - \text{sgn}(\frac{(z \bmod 100)}{10}) \cdot 10 - \text{sgn}((z \bmod 100) \bmod 10)$ (sgn steht hier für Signumfunktion; in diesem Schritt vergeht die Hälfte der durchschnittlichen Bearbeitungszeit)

Pro Zug wird entweder ein Student ausgewählt, der die nächste Aufgabe übernimmt, oder eine Zeiteinheit (in diesem Fall die Hälfte der durchschnittlichen Bearbeitungszeit) vorangeschritten. Damit die zweite Möglichkeit nur in Frage käme, wenn keine validen Belegungen erster Möglichkeit vorhanden sind, kann man die Aktionskosten der zweiten Möglichkeit entsprechend hoch setzen. So könnte man den Algorithmus auch terminieren lassen, indem man zusätzlich zulässige Höchstkosten definiert. Oder indem man z gegen 0 prüft und auf valide Belegungen erster Möglichkeit testet. In dieser Form terminiert der Algorithmus erst, wenn er eine valide Belegung für die a_i gefunden hat und $z = 0$.

1.b)

Der Verzweigungsgrad beträgt 4, da wir für die erste Aufgabe 3 Studenten auswählen, oder eine Zeiteinheit verstreichen lassen können.

Ohne andere wie in (1.a) beschriebenen Terminierungsformen ist die maximale Tiefe des Baumes unendlich, da man immer eine Zeiteinheit verstreichen lassen kann.

1.c)

Tiefensuche eignet sich nicht da die maximale Tiefe des Baumes unendlich beträgt. Breitensuche würde dagegen gleich eine optimale (d.h. geringste Tiefe) Lösung finden. Best-First-Search (als eigentlich informierte Suche) könnte mithilfe der in 1.b) vorgeschlagenen Aktionskosten sowohl eine optimale Lösung finden, als schneller zu dieser Lösung kommen.

1.d)

Es ist schwierig eine nicht-triviale Heuristik für dieses Problem zu finden, da es ein kombinatorisches Problem ist,

Aufgabe 2

2.a)