Prof. Dr. Rolf Wanka
Bernd Bassimir, Alexander Raß, Jonas Schreiner,
Stefan Schwanecke, Jeanette Wohnig

Erlangen, 4. Dezember 2018

Abgabe bis $\boxed{\text{11.12.18, 23:59 Uhr}}$

Übungen zur Vorlesung

Berechenbarkeit und Formale Sprachen

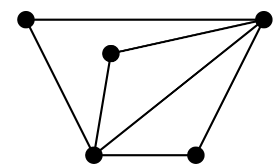
WS 2018/2019

Blatt 6

Je mehr Plus-Zeichen $+$ bei einer Aufgabe, desto wichtiger, je mehr Sterne $\star$, desto schwieriger.

AUFGABE 30:

[Präsenzaufgabe, $++,\star$] Sei $G = (V,E)$ ein ungerichteter Graph. Eine Teilmenge $U \subseteq V$ der Knotenmenge heißt unabhängige Menge (engl.: *independent set*), wenn es in der Kantenmenge E keine Kanten zwischen Knoten aus U gibt. Formal: $\forall u,v \in U : \{u,v\} \notin E$. (Finden Sie probehalber eine unabhängige Menge der Größe 3 im Graphen G rechts)



G

Das *Independent Set Problem* ist die Sprache

$$\text{IS} = \{\langle G,k\rangle \mid G \text{ ist ein ungerichteter Graph, } k \in \mathbb{N}, G \text{ hat eine unabhängige Menge } U \text{ mit } |U| = k\} \; .$$

(a) Geben Sie einen polynomiell beschränkten Verifizierer für IS an.

(b) Sei $G = (V,E)$ ein ungerichteter Graph. Nehmen Sie an, Sie könnten IS in Zeit $O(n^\ell)$, d. h. in Polynomzeit, mit der Turingmaschine $M_{\text{ent\_IS}}$ entscheiden.

  (i) Beschreiben Sie einen Algorithmus, der mit Hilfe von $M_{\text{ent\_IS}}$ die Größe einer größten unabhängigen Menge in G bestimmt. Welche Laufzeit hat er?

  (ii) Beschreiben Sie einen Algorithmus, der eine größte unabhängige Menge $U_{\max}$ berechnet. Welche Laufzeit hat er?

AUFGABE 31 (4 Punkte)**:**

[$+++,\star$]

(a) Zeigen Sie: Das Reduktionskonzept ist transitiv, d. h. es gilt: $L_1 \leq L_2$ und $L_2 \leq L_3 \Rightarrow L_1 \leq L_3$

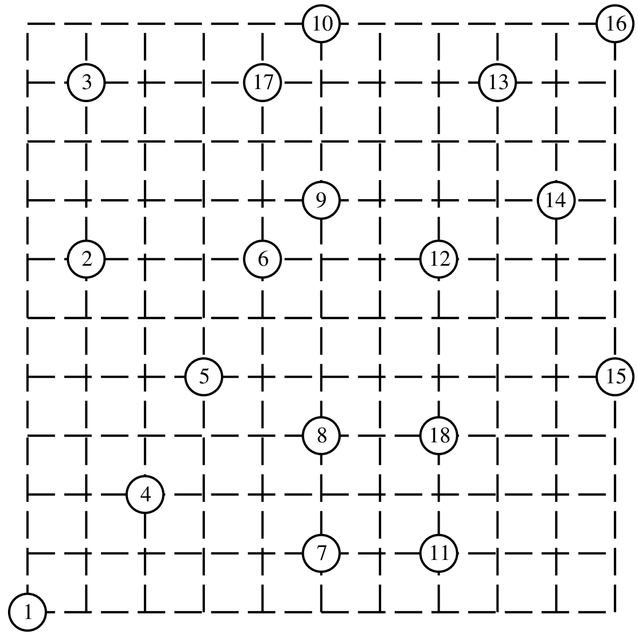
(b) Wenn die Laufzeit der Berechnung der ersten Reduktionsfunktion $t_1(n) = 11n^3$ ist und die der zweiten $t_2(n) = 42n^3 + 2n^2$, wie groß ist dann die Laufzeit der Berechnung Ihrer Reduktionsfunktion für $L_1 \leq L_3$ höchstens?

AUFGABE 32 (4 Punkte)**:**

[$+,\star\star$] Gegeben ist auf der nächsten Seite eine Leiterplatte der Größe 11×11 und darauf 18 markierte Stellen, in die jeweils ein Loch gebohrt werden soll. Ein Bohrer fährt, von zwei Motoren angetrieben, einer für die x-Richtung, einer für die y-Richtung, die markierten Stellen ab und bohrt jeweils das gewünschte Loch. Der Verschleiß an den beiden Motoren ist proportional zur Summe der Beträge der in x- und y-Richtung zurückgelegten Abstände. Z. B. ist der Abstand von Loch 4 zu Loch 5 (und umgekehrt) gleich 3 (*Manhattan-Norm*). Der Bohrer beginnt bei Loch 1, soll alle Löcher bohren und zum Ende wieder bei Loch 1 sein, um die nächste Leiterplatte bearbeiten zu können.

Finden Sie auf der gegebenen Leiterplatte drei verschiedene solche Bohrtouren mit geringem Verschleiß.

Hinweis: Kürzeste Touren haben die Länge 58. Zum Ausprobieren und für die Abgabe können Sie das Bild von unserer Webseite herunterladen.



AUFGABE 33 (4 Punkte)**:**
[++, $\star$] Zum Bohrproblem aus Aufgabe 32 gibt es insgesamt $\frac{1}{2} \cdot 17! = 177\,843\,714\,048\,000$ Touren. R. Borndörfer und M. Grötschel vom Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB) haben einmal *alle* Rundreisen konkret aufgezählt und analysiert, um eine Verteilung der Längen aller möglichen Touren zu sehen. Dafür haben sie auf schnellen Computern des Jahres 2007 fast 11 Tage reiner Rechenzeit benötigt. In dem Aufsatz

> Martin Grötschel. Schnelle Rundreisen: Das Travelling-Salesman-Problem. S. 95–129 in: Stephan Hußmann, Brigitte Lutz-Westphal (Hrsg.). *Kombinatorische Optimierung erleben*. Vieweg, 2007.
>
> `http://opus4.kobv.de/opus4-zib/files/890/ZR-05-57.pdf`

können Sie einiges zu diesen Arbeiten lesen.

(a) Nehmen Sie nun an, daß die Eingabe 30 Bohrlöcher enthält. Wie lange würde das Programm von Borndörfer und Grötschel brauchen, um auf den damaligen Rechnern alle möglichen Touren aufzuzählen und zu analysieren? Gehen Sie dabei davon aus, daß sich die Auswertungszeit einer einzelnen Tour nicht ändert.

(b) Gordon Moore hat 1965 einen berühmten Aufsatz geschrieben, in dem er das heute *Mooresches Gesetz* genannte Phänomen prognostizierte.

> G. E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38:114–117, 1965.

Unter `http://dx.doi.org/10.1109/JPROC.1998.658762` kann der bloß vier Seiten lange, klassische und sehr(!) lesenswerte Aufsatz heruntergeladen (nach „Download PDF“ suchen) werden (machen Sie sich auch mit der Bezeichnung doi vertraut).
Moore äußert sich nur über die Integrationsdichte von, wie wir heute sagen, VLSI-Chips. Allderings geht mit der von ihm vorhergesagten Verdopplung der Anzahl der Transitoren je IC im Rhythmus von zwei Jahren auch eine Steigerung der Rechenleistung von Computern einher. Deswegen geht man auch davon aus, daß sich ca. alle 24 Monate die von dann neuen Computern erreichte Rechengeschwindigkeit verdoppelt.
Wenn wir die Gültigkeit des Mooreschen Gesetzes voraussetzen, in welchem Jahr wird man die Aufzählung der 30-Löcher-Touren in 11 Tagen durchführen können?

AUFGABE 34 (4 **Bonus**- Punkte)**:**
Aus Aufgabe 23 auf Blatt 4 kennen Sie die Busy-Beaver-Funktion $\mathrm{BB}(n)$. Sie ist total, aber nicht berechenbar.

Sei $\mathcal{BB}_1 = \{(n, \mathrm{BB}(n)) \mid n \in \mathbb{N}\}$. Zeigen Sie, daß $\mathcal{BB}_1$ nicht rekursiv aufzählbar ist.

Hinweis: Eine Reduktion ist vermutlich nicht möglich. Aber Aufgabe 28 von Blatt 5 könnte sich für die Herleitung eines Widerspruchs als hilfreich erweisen. ☺