Radboud University Nijmegen

Science Faculty

Exam **Complexity IBC028** June 21, 2021, 8.30 – 10.30

The maximum number of points per question is given in the margin. (Maximum 100 points in total.)

When using well-known results that we have seen in the course, clearly state the result you are using; you don't have to prove it again.

---

(20) 1. We have a recursive algorithm whose time complexity $T(n)$ satisfies

$$T(n) = 7T(n-2) + 5T(n-3) + f(n),$$

with $f(n) = \Theta(n^3)$. Prove that $T(n) = \mathcal{O}(3^n)$.

Solution:

We prove $T(n) \leq c3^n$ by induction on n for n sufficiently large ($n \geq N$ for some N) and c to be chosen. We have $f(n) = \Theta(n^3)$, so we may assume we have an N_0 and $d > 0$ such that $f(n) \leq dn^3$ for $n \geq N_0$. Then we have, for $n \geq N_0$,

$$\begin{array}{rcl} T(n) & = & 7T(n-2) + 5T(n-3) + f(n) \\ & \stackrel{IH}{\leq} & 7c3^{n-2} + 5c3^{n-3} + dn^3 \\ & = & \frac{7}{9}c3^n + \frac{5}{27}c3^n + dn^3 \\ & = & \frac{26}{27}c3^n + dn^3 \end{array}$$

So $T(n) \leq \frac{26}{27}c3^n + dn^3$. For n sufficiently large (and larger than N_0), we have $dn^3 \leq \frac{1}{27}c3^n$ (because $\frac{n^3}{3^n} \to 0$ for $n \to \infty$ and so $\frac{n^3}{3^n} \leq \frac{c}{27d}$ for n sufficiently large). So we choose $c > 0$ arbitrarily, say $c := 1$, and we have $T(n) \leq c3^n$ for n sufficiently large, so $T(n) = \mathcal{O}(3^n)$.

(20) 2. We have a recursive algorithm that, on an input of size n, does $3i$ recursive calls on input of size $\frac{n}{3}$ plus additional computations of time complexity $\Theta(n^2)$. Determine the time complexity of this algorithm for $i = 1, 2, 3, 4$.

Solution:

We have $T(n) = 3iT(\frac{n}{3}) + f(n)$ with $f(n) = \Theta(n^2)$. Following the Master theorem (MT) we get the following: $b = 3$ in all cases, but a varies:

$i=1$ Then $a=3$, so $D=\log_b a=\log_3 3=1$, so $f(n)=\Omega(n^{D+\epsilon})$ for some $\epsilon>0$, so we are in (case 3) of the MT. We check the side condition: $af(\frac{n}{b})\leq cf(n)$ for some $c<1$ for n sufficiently large. This holds for $c=1/3$: $3(\frac{n}{3})^2\leq\frac{1}{3}n^2$.
So $T_1=\Theta(n^2)$.

$i=2$ Then $a=6$, so $D=\log_b a=\log_3 6<2$, so $f(n)=\Omega(n^{D+\epsilon})$ for some $\epsilon>0$, so we are in (case 3) of the MT. We check the side condition: $af(\frac{n}{b})\leq cf(n)$ for some $c<1$ for n sufficiently large. This holds for $c=\frac{1}{12}$: $3(\frac{n}{6})^2\leq\frac{1}{12}n^2$.
So $T_2=\Theta(n^2)$.

$i=3$ Then $a=9$, so $D=\log_b a=\log_3 9=2$, so $f(n)=\Theta(n^D)$, so we are in (case 2) of the MT.
So $T_3=\Theta(n^2\log n)$.

$i=4$ Then $a=12$, so $D=\log_b a=\log_3 12>2$, so $f(n)=\mathcal{O}(n^{D-\epsilon})$ for some $\epsilon>0$, so we are in (case 1) of the MT.
So $\Theta(n^{\log_3 12})$.

This solves all cases.

3. Suppose we have two algorithms A_1 and A_2 for which we have bounds on the running time, given by T_1 and T_2, respectively for which we know the following (for some constants c and d).

$$
\begin{aligned}
T_1(n) &= T_1(\left\lfloor\frac{n}{7}\right\rfloor)+T_1(\left\lfloor\frac{2n}{7}\right\rfloor)+T_1(\left\lfloor\frac{3n}{7}\right\rfloor)+cn\\
T_2(n) &= T_2(\left\lfloor\frac{n}{2}\right\rfloor)+T_2(\left\lfloor\frac{n}{3}\right\rfloor)+T_2(\left\lfloor\frac{n}{6}\right\rfloor)+dn
\end{aligned}
$$

(10) (a) Use the recursion tree method to compute an f_1 such that algorithm A_1 is $\Theta(f_1(n))$. (Subtleties due to rounding may be ignored.)

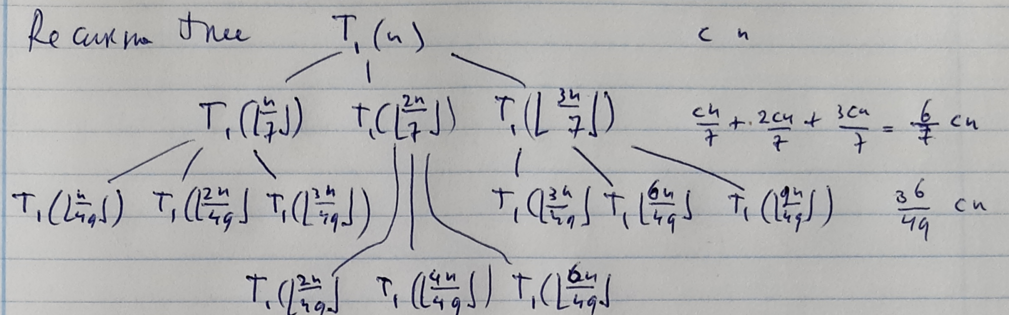
(10) (b) Use the recursion tree method to compute an f_2 such that algorithm A_2 is $\Theta(f_2(n))$. (Subtleties due to rounding may be ignored.)

Solution:
T_1 is $\Theta(n)$, T_2 is $\Theta(n\log n)$:

Exam 2021

③ $T_1(n) = T_1\left(\left\lfloor \frac{n}{7} \right\rfloor\right) + T_1\left(\left\lfloor \frac{2n}{7} \right\rfloor\right) + T_1\left(\left\lfloor \frac{3n}{7} \right\rfloor\right) + cn$

Recursion tree



⋮

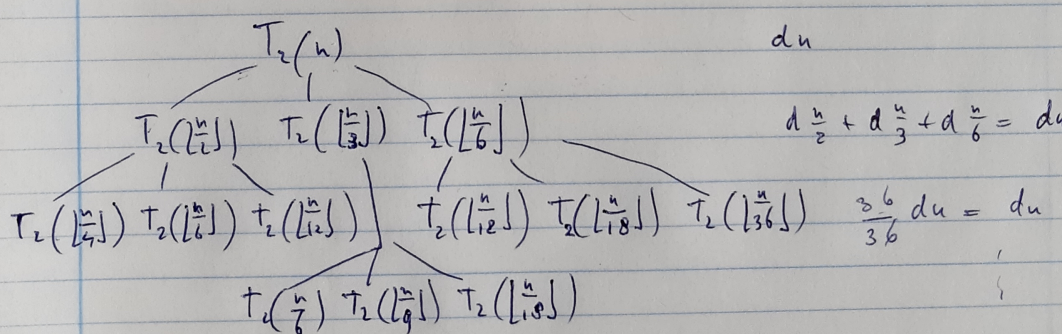
We obtain:

- per level a contribution of $\left(\frac{6}{7}\right)^i cn$
- depth: $\log_7 n$

So $T_1(n) \simeq \sum_{i=0}^{\log_7 n} \left(\frac{6}{7}\right)^i cn \simeq cn \cdot \sum_{i=0}^{\infty} \left(\frac{6}{7}\right)^i = cn \cdot \frac{1}{1-\frac{6}{7}} = 7cn$

So $T_1(n) = \Theta(n)$

$T_2(n) = T_2\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T_2\left(\left\lfloor \frac{n}{3} \right\rfloor\right) + T_2\left(\left\lfloor \frac{n}{6} \right\rfloor\right) + dn$



We obtain:

- per level a contribution of dn
- depth: $\log_6 n$

So $T_2(n) = \sum_{i=0}^{\log_6 n} dn = dn \log_6 n$

So $T_2(n) = \Theta(n \log n)$

See next page

4. We have defined the problem *not-all-equal*-3CNF-SAT, Neq3CNF-SAT(φ), as the problem of deciding for a formula $\varphi \in$ 3CNF whether there is an assignment such that in every clause in φ, **at least one literal is true and at least one literal is false**. Similarly, we have Neq4CNF-SAT: the problem of deciding for a formula $\varphi \in$ 4CNF whether there is an assignment such that in every clause in φ, **at least one literal is true and at least one literal is false**.

(10) (a) Describe a procedure to transform a disjunction of 4 literals $\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4$ into a 3CNF, φ, such that

$$\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4 \text{ is Neq4-satisfiable if and only if } \varphi \text{ is Neq3-satisfiable.}$$

Prove that your procedure satisfies this property.

(10) (b) It is given that Neq4CNF-SAT is NP-complete. Prove that Neq3CNF-SAT is NP-complete.

Solution:

(a) Send $\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4$ to $(\ell_1 \vee \ell_2 \vee a) \wedge (\ell_3 \vee \ell_4 \vee \neg a)$ for a a fresh atom. We have:

$\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4$ is Neq4-satisfiable if and only if $(\ell_1 \vee \ell_2 \vee a) \wedge (\ell_3 \vee \ell_4 \vee \neg a)$ is Neq3-satisfiable.

Proof:

($\Rightarrow$): Suppose v is an Neq4-valuation for $\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4$. There are several cases.

- $v(\ell_1) = v(\ell_2) = 1$. Then set $v(a) = 0$, then v is an Neq3-valuation for $\ell_1 \vee \ell_2 \vee a$ and $v(\neg a) = 1$ and (at least) one of ℓ_3, ℓ_4 has $v(\ell_i) = 0$, so v is an Neq3-valuation for $\ell_3 \vee \ell_4 \vee \neg a$ as well.
- $v(\ell_1) = 0$, $v(\ell_2) = 1$. Then no matter what $v(a)$ is, v is an Neq3-valuation for $\ell_1 \vee \ell_2 \vee a$. Set $v(a)$ to 0 or 1, depending on $v(\ell_3)$ and $v(\ell_4)$ to make sure that v is an Neq3-valuation for $\ell_3 \vee \ell_4 \vee \neg a$.
- $v(\ell_1) = v(\ell_2) = 0$. (This is the "mirror case" of the first.) Then set $v(a) = 1$, then v is an Neq3-valuation for $\ell_1 \vee \ell_2 \vee a$ and $v(\neg a) = 0$ and (at least) one of ℓ_3, ℓ_4 has $v(\ell_i) = 1$, so v is an Neq3-valuation for $\ell_3 \vee \ell_4 \vee \neg a$ as well.

($\Leftarrow$): Suppose v is an Neq3-valuation for $(\ell_1 \vee \ell_2 \vee a) \wedge (\ell_3 \vee \ell_4 \vee \neg a)$. There are two cases.

- $v(a) = 1$. Then $v(\ell_1) = 0$ or $v(\ell_2) = 0$. Also $v(\neg a) = 0$, so $v(\ell_3) = 1$ or $v(\ell_4) = 1$. So v is an Neq4-valuation for $\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4$.
- $v(a) = 0$. Then $v(\ell_1) = 1$ or $v(\ell_2) = 1$. Also $v(\neg a) = 1$, so $v(\ell_3) = 0$ or $v(\ell_4) = 0$. So v is an Neq4-valuation for $\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4$.

(b) That Neq3CNF-SAT is in NP follows from the fact that there is a certificate that can easily be checked to be Neq3-satisfiable in polynomial time:

- The certificate is the assignment $v : \text{Atom} \to \{0, 1\}$.
- Checking the certificate means that we have to check that v is a Neq3 valuation for a 3CNF, φ. That means we have to check for each clause $\ell_1 \vee \ell_2 \vee \ell_3$ that at least one of the literals becomes true and at least one of the literals becomes false under v. This can easily be checked in polynomial time for a given φ. (Even linear time.)

That Neq3CNF-SAT is NP-Hard is proven by polynomially reducing Neq4CNF-SAT to Neq3CNF-SAT: Neq4CNF-SAT $\leq_P$ Neq3CNF-SAT.

The reduction from Neq4CNF-SAT to Neq3CNF-SAT is given by the polynomial function f defined as follows.

$$f(\bigwedge_{i=1}^{n} C_i) = \bigwedge_{i=1}^{n} f(C_i)$$

where $f(\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4) = (\ell_1 \vee \ell_2 \vee a) \wedge (\ell_3 \vee \ell_4 \vee \neg a)$ for a a fresh atom (for every clause C_i a new fresh atom), the procedure indicated in part (a). Then φ is Neq4-satisfiable iff $f(\varphi)$ is Neq3-satisfiable, as has been shown in part (a).

5. Define, for $G = (V, E)$ an undirected graph, the problem "relaxed 3Color", r3Color(G), as the problem to decide whether G can be 3-colored where **at most one edge can have both endpoints of the same color** and each other edge has two endpoints with a different color.

(5) (a) Draw a graph that can be "relaxed-3-colored", but not 3-colored.

(15) (b) Prove that r3Color is NP-complete.

Hint Use the NP-hardness of 3Color; add a simple graph to your graph.

Solution:

(a) The simplest graph consists of 4 vertices, a_1, a_2, a_3, a_4 that are all connected with eachother via edges. So, a tetrahedron. Observe that any 3-coloring of a_1, a_2, a_3, a_4 has at least 1 edge where the end points have the same color, so it cannot be 3-colored, but if two edges are colored the same, it is relaxed-3-colored.

(b) That $r3\textsf{Color}$ is in NP follows from the fact that there is a certificate that can easily be checked in polynomial time:

- The certificate is the coloring of the vertices $c : V \to \{r, y, g\}$.
- Checking the certificate means that we have to check that c is a relaxed-3-coloring of (V, E). That means we have to check for each edge $(v, u) \in E$ whether $c(v) = c(u)$. There can be at most one edge where $(v, u) \in E$ whether $c(v) = c(u)$. This can easily be checked in polynomial time. (Even linear time.)

That $r3\textsf{Color}$ is NP-Hard is proven by polynomially reducing: $3\textsf{Color} \leq_P r3\textsf{Color}$. The definition of the reducing map f is: $f(G)$ is G with the tetrahedron added (disconnected from the rest of G, or possibly with one vertex shared.) We need to prove that (V, E) is 3-colorable if and only if $f(V, E)$ is relaed-3-colorable.

$\Rightarrow$: If (V, E) is 3-colorable, then $f(V, E)$ is relaxed-3-colorable, by coloring the tetrahedron with 3 colors and one edge having two endpoints with the same color.

$\Leftarrow$: If $f(V, E)$ is relaxed-3-colorable, then in the tetrahedron one edge has two endpoints with the same color. So the "rest of $f(V, E)$" is 3-colorable, but that's just (V, E). So (V, E) is 3-colorable.

END