Complexiteit

Uitwerkingen opgave 43 + 47 + 48c

opgave 43
Te bewijzen: minstens $c*n*n$ vergelijkingen (zekere c) in de worst case, ofwel te bewijzen dat er een invoer is waarop Shellsort ten minste $c*n*n$ vergelijkingen doet.
Neem een rijtje getallen waarbij de KLEINSTE $n/5$ getallen in oplopende volgorde op de array-posities met array-indices 5, 10, 15, 20, 25, ... staan. Dankzij de keuze van de stapgroottes (altijd 5-machten) worden de elementen op posities 5, 10, 15, 20, 25, ... alleen onderling vergeleken en blijven deze getallen tot aan de laatste ronde (met stapgrootte 1) op hun plaats staan omdat ze reeds gesorteerd zijn. Er worden in die rondes dus wel vergelijkingen gedaan, en mogelijk ook elementen op andere posities onderling verwisseld, maar de getallen 1 t/m $\frac{n}{5}$ blijven staan. Pas in de laatste doorgang worden ze verplaatst, en wel op hun uiteindelijke postitie gezet via insertion sort. Het aantal vergelijkingen in deze ronde is minstens het aantal vergelijkingen nodig om de $n/5$ kleinste getallen op hun juiste posities te zetten. Het getal op positie 5 moet naar plek 1, het getal op plek 10 naar 2, het getal op plek 15 naar 3, ..., het getal op plek $5*i$ naar plek i, etcetera. Dat geeft totaal $4+9+13+...+4*i+1+...4*\frac{n}{5}+1$ vergelijkingen, en dat is groter dan $4+8+12+...+4*\frac{n}{5}=4*(1+2+3+...+\frac{n}{5})=2*\frac{n}{5}*(\frac{n}{5}+1)$ vergelijkingen. Het totaal aantal vergelijkingen is ten minste dit aantal, dus $\in\Omega(n^2)$.

Opmerking: je kunt de afschatting ook (grover, maar eenvoudiger) als volgt maken:
Het getal op positie 5 moet naar plek 1, het getal op plek 10 naar 2, het getal op plek 15 naar 3,..., etcetera. Dat geeft totaal $4+9+13+...$ (voor de $n/5$ kleinste getallen dus), en dat kun je als volgt afschatten: $\geq 1+2+3+...+n/5$ (want $n/5$ termen), en dat is weer gelijk aan: $\frac{1}{2}*\frac{n}{5}*(\frac{n}{5}+1)=\Omega(n^2)$.

opgave 47
a.$[(x^4+2)*[(x^2+4)*[x+6]+[x+-20]]+[(x^2+-10)*[x+-10]+[x+-107]]]$
b. Oplossen van de recurrente betrekking voor $A(k)$.

(1) Herhaald invullen
$A(k)=2\,A(k-1)+2=2\,(2\,A(k-2)+2)+2=2^2\,A(k-2)+2^2+2=2^2\,(2\,A(k-3)+2)+2^2+2=2^3\,A(k-3)+2^3+2^2+2=\ldots=2^{k-1}\,A(1)+2^{k-1}+2^{k-2}+\ldots+2^2+2=2^{k-1}+\sum_{i=0}^{k-1}2^i-1=2^{k-1}-1+2^k-1=3\cdot 2^{k-1}-2$.

(2) Bewijs met inductie dat de in (1) gevonden functie inderdaad voor alle $k\geq 1$ de oplossing (dus $A(k)$) is van de recurrente betrekking:
. Voor $k=1$ klopt het: $3\cdot 2^0-2=1=A(1)$.
. Stel dat inderdaad $A(k)=3\cdot 2^{k-1}-2$ voor alle $k\leq K-1$. Nu aantonen dat het dan ook klopt voor $k=K$.
$A(K)=2\,A(K-1)+2=$ (inductie-aanname) $2\cdot(3\cdot 2^{K-2}-2)+2=3\cdot 2^{K-1}-2$. Klopt.
Uitgedrukt in n is dit: $\frac{1}{2}(3n-1)$.

c. Totaal aantal vermenigvuldigingen uitgedrukt in n: $\frac{n-1}{2}+\lg(n+1)-1$. Dit is niet in tegenspraak met de afgeleide ondergrens van n vermenigvuldigingen omdat het hier gaat om polynomen die in een zeer speciale vorm gebracht zijn. Het in de juiste vorm brengen kost ook een zeker aantal +/- en * operaties. Het totaal aantal, dus inclusief het preprocessen, is hier de juist maat voor de complexiteit. Dat aantal moet met de n uit de stelling vergeleken worden. Alleen wanneer het polynoom heel vaak in verschillende punten moet worden geevalueerd heeft het zin om preprocessing te doen. Dit hoeft namelijk maar één keer, waarna dan herhaald de snellere evaluatie kan worden toegepast.

opgave 48
a. Zie ook 47 a.
c. Recurrente betrekking voor $A(k)$:
$A(k) = 1 + 2^{k-1} + 2\,A(k-1)$; $A(1) = 0$.

Oplossen $A(k)$ door herhaald invullen:
$A(k) = 1+2^{k-1}+2\,A(k-1) = 1+2^{k-1}+2(1+2^{k-2}+2\,A(k-2) = 1+2+2\cdot 2^{k-1}+2^2\,A(k-2) = \ldots = (1+2+\ldots+2^{k-2}) + (k-1)\,2^{k-1} + 2^{k-1}\,A(1) = 2^{k-1} - 1 + (k-1)\,2^{k-1} = k \cdot 2^{k-1} - 1$.

Met inductie bewijzen dat $k \cdot 2^{k-1} - 1$ de oplossing is:
. Het klopt voor $k = 1$: $1 \cdot 2^0 - 1 = 0 = A(1)$.
. Stel dat $k \cdot 2^{k-1} - 1$ voor $k \leq K-1$ inderdaad de oplossing is van de recurrente betrekking, dus dat $A(k) = k \cdot 2^{k-1} - 1$ voor alle $k \leq K-1$. Nu aantonen dat het dan ook klopt voor $k = K$.
$A(K) = 1 + 2^{K-1} + 2\,A(K-1) = 1 + 2^{K-1} + 2((K-1) \cdot 2^{K-2} - 1) = 1 + 2^{K-1} + (K-1) \cdot 2^{K-1} - 2 = K \cdot 2^{K-1} - 1$.

Uitgedrukt in n betekent dit dat het aantal optellingen gelijk is aan: $\frac{1}{2}(n+1)\lg(n+1) - 1$.
Immers: $n = 2^k - 1 \Longrightarrow 2^k = n + 1$ en $k = \lg(n+1)$.

Recurrente betrekking voor $V(k)$ (het aantal vermenigvuldigingen):
$V(k) = 2^{k-1} + 2\,V(k-1)$; $V(1) = 0$.

Vinden/bewijzen van de oplossing hiervan is analoog aan dat voor $A(k)$. Hieronder is alleen het vinden van de oplossing weergegeven.

$V(k) = 2^{k-1} + 2\,V(k-1) = 2^{k-1} + 2(2^{k-2} + 2\,V(k-2)) = 2 \cdot 2^{k-1} + 2^2\,V(k-2) = \ldots = (k-1)\,2^{k-1} + 2^{k-1}\,V(1) = (k-1)\,2^{k-1}$.

Dus het aantal vermenigvuldigingen uitgedrukt in n: $\frac{1}{2}(n+1)(\lg(n+1)-1)$. Merk op dat het aantal vermenigvuldigingen uit de preprocessing-stap en het aantal vermenigvuldigingen uit de evaluatie (opgave 47) samen $\frac{1}{2}(n+1)(\lg(n+1) - 1) + \frac{n-1}{2} + \lg(n+1) - 1 \geq n + 1$ (als $n \geq 3$) bedraagt.