APC Special Assignment 1

Nicolas Muntwyler - 17-915-737 - Gruppe C

Oktober 2019

Exercise 1

a) $E[D_1(0)] = 0$
Only possibility for T with n=0 is:

$$[-\infty, \infty)$$

Which has a Depth of left most leaf of 0.

$E[D_1(1)] = 1$
Only possibility for T with n=1 is:

$$\begin{tikzcd} & {[-\infty, \infty)} \arrow[dl] \arrow[dr] & \\ {[-\infty, a)} & & {[a, \infty)} \end{tikzcd}$$

Which has a Depth of left most leaf of 1.

b) If we know that the first element inserted in a_k, the tree must look like this:

$$\begin{tikzcd} & {[-\infty, \infty)} \arrow[dl] \arrow[dr] & \\ {[-\infty, a_k)} & & {[a_k, \infty)} \end{tikzcd}$$

So when we insert $p_2, .., p_n$ every $p_i \geq a_k$ goes to the right child of the root (i.e $[a_k, \infty)$) Hence for $D_1(n)$ only the element $p_i < a_k$ matter since only these will land in the left subtree. Therefore we can write :

$$E[D_1(n)|p_1 = a_k] = 1 + \text{Expected Depth of left most leaf in left subtree} \tag{1}$$

Wlog we assume that $P \in [n]$. So all elements $p_i < a_k$ are $\in [k-1]$ Hence the expected Depth of the left most leaf in the left subtree is just:

$$E[D_1(k-1)] \tag{2}$$

From (1) and (2) we get:

$$E[D_1(n)|p1 = a_k] = 1 + E[D_1(k-1)]$$

c) wlog $P \in [n]$

$$
\begin{aligned}
E[D_1(n)] &= \sum_{k=1}^{n} \underbrace{E[D_1(n)|p_1 = a_k]}_{\begin{cases} 1 & \text{if k = 1} \\ 1 + E[D_1(k-1)] & k \geq 2 \end{cases}} \cdot \underbrace{Pr[p_1 = a_k]}_{1/n} \\
&= \frac{1}{n} \cdot \sum_{k=2}^{n} (1 + E[D_1(k-1)]) + \frac{1}{n}
\end{aligned} \tag{3}
$$

d) Let:

$$x_n := E[D_1(n)]$$

Then we get for $n \geq 3$:

$$x_n = \frac{1}{n} + \sum_{k=2}^{n} (1 + x_{k-1}) \frac{1}{n} = \frac{1}{n} + \frac{1}{n} \sum_{k=2}^{n} (1 + x_{k-1})$$

$$x_{n-1} = \frac{1}{n-1} + \frac{1}{n-1} \sum_{k=2}^{n-1} (1 + x_{k-1})$$

Which can be transform to (for $n \geq 3$):

$$x_n n = 1 + \sum_{k=2}^{n} (1 + x_{k-1})$$

$$x_{n-1}(n-1) = 1 + \sum_{k=2}^{n-1} (1 + x_{k-1})$$

By subtracting the two identities we get (for $n \geq 3$):

$$
\begin{aligned}
nx_n - (n-1)x_{n-1} &= 1 + x_{n-1} \\
&= nx_n - nx_{n-1} + x_{n-1} \\
&= 1 + x_{n-1} \\
&= nx_1 - nx_{n-1} = 1 \\
&= nx_1 = 1 + nx_{n-1} \\
&= x_n = \frac{1}{n} + x_{n-1} \\
&= x_n = \frac{1}{n} + x_{n-1}
\end{aligned} \tag{4}
$$

Together with $x_2 = 1.5$, successive invocation yields:

$$
\begin{aligned}
x_n &= \frac{1}{n} + x_{n-1} = \frac{1}{n} + \frac{1}{n-1} + x_{n-2} = ... \\
&= \frac{1}{n} + \frac{1}{n-1} + ... + \frac{1}{3} + \underbrace{x_2}_{1.5} = H_n
\end{aligned} \tag{5}
$$

Lets prove it by induction:
To prove: $x_n = \sum_{i=1}^{n} \frac{1}{i}$
Base (n=2): $x_2 = 1.5 = \frac{1}{2} + 1 = \sum_{i=1}^{2}$ (see below for the 1.5)
Induction Hypothesis: $x_n = \sum_{i=1}^{n} \frac{1}{n}$
Induction step: $x_{n+1} = \frac{1}{n+1} + x_n \overset{\text{I.H}}{=} \frac{1}{n+1} + \sum_{i=1}^{n} \frac{1}{i} = \sum_{i=1}^{n+1} \frac{1}{i}$

In order to calculate x_2 (wlog $a_k \in [n]$):

$$x_2 = E[D_1(2) = E[D_1(2)|p_1 = 1]\frac{1}{2} + E[D_1(2)|p_1 = 2]\frac{1}{2} = 1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{2} = 1.5$$

Where $E[D_1(2)|p_1 = 1] = 1$ since the binary tree has to look like this:

$$\begin{tikzcd}
& {[-\infty, \infty)} \arrow[dl] \arrow[dr] & & \\
{[-\infty, 1)} & & {[1, \infty)} \arrow[dl] \arrow[dr] & \\
& {[1, 2)} & & {[2, \infty)}
\end{tikzcd}$$

Where $E[D_1(2)|p_1 = 2] = 2$ since the binary tree has to look like this:

$$\begin{tikzcd}
& & {[-\infty, \infty)} \arrow[dl] \arrow[dr] & \\
& {[-\infty, 2)} \arrow[dl] \arrow[dr] & & {[2, \infty)} \\
{[-\infty, 1)} & & {[1, 2)} &
\end{tikzcd}$$

In conclusion we get: $E[D_1(n)] = H_n$

Exercise 2

a) We first define:

$$x_i := \begin{cases} 1 & \text{if insertion of } p_i \text{ changes nearest neighbour to x} \\ 0 & \text{otherwise} \end{cases}$$

$$\text{X} := \text{number of times nearest neighbour to x changes } = \sum_{i=1}^{n} x_i$$

And remark that we label the points in S from 1 to n, so that when two points share the same distance from x, we order them according to their labels. So if there exist two "nearest" neighbours to x, we choose the one with the lower label. That way we have a strict order and we always have unique nearest neighbours. This change results in only more nearest neighbour changes. So the expected number of nearest neighbour changes in our new problem is an upper bound for the original problem. So if we have that the new problem has an expected number of nearest neighbour changes of $O(\log n)$ (as we will see), then also the expected number of nearest neighbour changes in the original problem is also at most $O(\log n)$

We then note, that:
Prove nearest neighbour changes $O(\log n)$ with probability at least $1 - \frac{1}{n^6}$
$\iff$
Pr[X$\leq c_2 \cdot \log n] \geq 1 - \frac{1}{n^6}$ (where c_2 is a constant)

Since X is the sum of independent Bernoulli random variables, we can use the Chernoff bound:

$$Pr[X \geq (1+\delta)E[X]] \leq e^{-\frac{E[X]\delta^2}{3}}$$

$$1 - Pr[X \geq (1+\delta)E[X]] \geq 1 - e^{-\frac{E[X]\delta^2}{3}}$$

$$Pr[X \leq (1+\delta)E[X]] \geq 1 - e^{-\frac{E[X]\delta^2}{3}}$$

After exercise 3 in KW42 the expected number of distinct nearest neighbors is $H_n = O(\log n)$ Hence $E[X] \leq c_1 \cdot \log n$ (c_1 has to exist). We can use this to get:

$$Pr[X \leq (1+\delta)c_1 \cdot \log n] \geq 1 - e^{-\frac{c_1 \cdot \log n \delta^2}{3}}$$

We can choose $\delta = \frac{1}{2}$ to get:

$$Pr[X \leq 1.5c_1 \cdot \log n] \geq 1 - e^{-\log n \frac{c_1}{12}}$$

$$Pr[X \leq 1.5c_1 \cdot \log n] \geq 1 - (e^{\log n})^{-\frac{c_1}{12}}$$

$$Pr[X \leq 1.5c_1 \cdot \log n] \geq 1 - n^{-\frac{c_1}{12}}$$

$$Pr[X \leq 1.5c_1 \cdot \log n] \geq 1 - \frac{1}{n^{\frac{c_1}{12}}}$$

We choose $c_1 \geq 72$ to have $1 - \frac{1}{n^{\frac{c_1}{12}}} \geq 1 - \frac{1}{n^6}$. That way we get:

$$Pr[X \leq 1.5c_1 \cdot \log n] \geq 1 - \frac{1}{n^6}$$

where $c_1 \geq 72$. But now $1.5c_1$ is just another constant c_2. With $c_2 \geq 1.5c_1$. Hence we can rewrite to:

$$Pr[X \leq c_2 \cdot \log n] \geq 1 - \frac{1}{n^6}$$

where $c_2 \geq 1.5c_1$ (hence a constant). This is the same as saying:
x changes the nearest neighbour $O(\log n)$ times, with probability bigger than $1 - \frac{1}{n^6}$.
Which was the statement to be proven.

b) We fist define

$$X := \text{number of such sequences on S over all points x in the plane}$$

We first note that proving that $O(n^4)$ such Sequences (as described in the exercise) on S over all points x in the plane exist, is the same as proving that $X \leq c_1 n^4$ for some $c_1 \in R$, for all n
We also note, that when we speak about Sequences, we assume its a sequence of the form described in the exercise (ordering of distance to neighbours).
Secondly we create $O(n^2)$ lines from our n point with $1 \leq i < j \leq n$:

$$l_{ij} = \text{Perpendicular bisector between point i and j}$$

There are $\binom{n}{2} = O(n^2)$ such lines.
These lines create an arrangement that has $O(n^4)$ faces (Fact given in Hint). Hence the number of faces is $\leq c_2 n^4$ for some $c_2 \in R$

Intuitively we know, that all points in the same face have the same sequence. And since only $O(n^4)$ such faces exist we can therefore only have $O(n^4)$ Sequences. But lets do it formally:

Claim: Number of these faces $\geq$ X

If the Claim holds we conclude our proof, because:

$$X \leq \text{Number of faces} \leq c_2 n^4$$

Proof of Claim

We want to argue, that every possible such sequence is represented by a face, which would prove our claim.

Let's postpone proving that for a bit and do some Observations:

(1) We note that each line l_{ij} creates an ordering:
We say that all points that lie on the same side as p_i are closer to p_i than to p_j. Vice versa we say that all points that lie on the same side as p_j are closer to p_j than to p_i. If a point is on the line we say it lies on the side where the point (either p_i or p_j) has the lower label.
Therefore for a point x that lies on the side of p_j of l_{ij}, we get the Sequence $S_x = (..., j, ..., i, ...)$.
Similarly for a point x that lies on the side of p_i of l_{ij} we get the Sequence $S_x = (..., i, ..., j, ...)$.

(2) For a given point $x \in R$, we get a strict ordering from all lines. There exists for each pair of points in S a bisector-line. And this line gives you an ordering. So in the end there is only one possible order/Sequence for this point.

(3) All points who have the same lines above/below/left/right, have the same ordering, or rather, the same Sequence S_x. The union of these points for the particular Sequence S_x is exactly a face. Hence we can associate this face with S_x. That way each face is associated with a Sequence. We also note that each face has a different Sequence associated, since every face is separated by at least one line l_{kl} from another face. And exactly this line gets us a different Sequence from the other face. In one face we have: $S_=(..., k, ..., l, ...)$ and in the other: $S = (..., l, ..., k, ...)$. This holds for any pair of faces.

Now we want to prove it:
Intuitively the face that contains the point x, which has the Sequence S_x, will have the same Sequence S_x associated with it. And since the union of all faces span the whole space R^2, each point x can be located to a face and hence each Sequence can be associated to a face.
But let's do it formally. We do a proof by contradiction:
Let S_x be a Sequence, that has no face associated with it. Then the face f (with associated Sequence S_f), in which x is located, must have a different Sequence than S_x. Let S_x and S_f agree on the first i points, but at the (i+1)-th point S_x has p_l and S_f has p_k ($p_k \neq p_l$). We now prove that this is not possible:
Since S_x and S_f agree on the fist i points and no point appears twice in the Sequence, we must have that p_k comes after p_l in S_x. Hence the Distance from x to p_l is smaller than the distance from x to p_k. Hence if we have a line l that is the bisector of p_k and p_l, x would have to lie on the same side as p_l. But then for the face, which contains x, we enforced the ordering $S_f = (..., l, ..., k, ...)$, since x is on the side of p_l. Hence we have that p_k comes after p_l in S_f. But this is a contradiction. Because if S_x and S_f are the same until the (i+1)-th point and S_x has p_l at the (i+1)-th point, we know that p_l isn't in the first i points of S_f. At the (i+1)th point in S_f, we have p_k. Hence p_l has to come after p_k in S_f.

c) The number of nearest-neighbour-changes for a (fixed) point, depends on the ordering of its neighbours. If two points have the same ordering of nearest neighbours, they will also have the same number of nearest-neighbour-changes. (Proof: Let a and b be points with the same sequence. The nearest neighbour changes for point a iff a point ($\in S$) got inserted, that has a lower place in the Sequence associated with point a. But then it also changes for b since b has the same Sequence. Vice versa the same argument holds.)
From b) we know, that there exist only $O(n^4)$ such orderings. Hence n^5 is an upper bound on the number of such orderings (for a sufficiently large n). For any of these orderings we could have, that for a fixed point of that ordering, the nearest neighbour changes more than $O(\log n)$ times. But this happens only with a probability of $O(\frac{1}{n^6})$ as we know from a). So we have n^5 chances to have a point that changes its nearest neighbour more than $O(\log n)$ times, each time with a probability of $\frac{1}{n^6}$. So the probability that the nearest neighbour changes more than $O(\log n)$ times for any of these orderings

is at most:

$$n^5 \cdot \frac{1}{n^6} = \frac{1}{n}$$

Therefore the probability, that it doesn't happen for any sequence is at least:

$$1 - \frac{1}{n}$$

Hence the probability, that every point in the plane changes its nearest neighbour $O(\log n)$ times is at least $1 - \frac{1}{n}$. Which is exactly what we wanted to prove.

Exercise 3

a) (i) We add together the number of comparisons in each step:

step 1: 0 comparisons
step 2: #Block · (#Comparisons in each Block) $= \frac{n}{\beta} \cdot (\beta \cdot \beta) = n\beta$
step 3: 0 comparisons
step 4: nD

Which totals to: $O(0 + n\beta + 0 + nD) = O(n\beta + nD)$

(ii) For steps 1-3 the comparisons stay the same. However in step 4 we have now the additional information that:

$$W = D \leq c \frac{n}{\sqrt{\beta}} \log(n)$$

We plug it in our in (i) derived formula to get:

$$O(n\beta + n \cdot nc \frac{n}{\sqrt{\beta}} \log(n)) = O(n\beta + n^2 \frac{1}{\sqrt{\beta}} \log(n))$$

Now lets comment on the probability. We want to prove, that this algorithm has a probability at least $1 - \frac{1}{n}$ to achieve $O(\log n)$ maximum dislocation:
In step 4 we apply A on P'. For W we choose $c \frac{n}{\sqrt{\beta}} \log(n)$. A with input W return with probability $1 - \frac{1}{n^8}$ a result that has $O(\log n)$ maximum dislocation, but only if the chosen W is $\geq$ D. (Which in our case is the case to a probability of $\geq 1 - \frac{1}{n^8}$). So in order to achieve that A (from step 4) returns us a result with $O(\log n)$ maximum dislocation, we need that $W \geq D$ on P'. Hence we get(for sufficiently large n):

$$\begin{aligned}
&\Pr[\text{Four-step-Algo returns } O(\log n) \text{ maximum dislocation}] \\
&= \Pr[\text{chosen W} \geq \text{D (at P')}] \cdot \Pr[\text{A returns } O(\log n) \text{ maximum dislocation} \mid \text{W} \geq \text{D (at P')}] \\
&\geq (1 - \frac{1}{n^7}) \cdot (1 - \frac{1}{n^8}) \\
&\geq 1 - \frac{1}{n^6} \\
&\geq 1 - \frac{1}{n}
\end{aligned}$$

(iii) We know:

$$O(n\beta + n^2 \frac{1}{\sqrt{\beta}} \log(n)) = \tilde{O}(n\beta + n^2 \frac{1}{\sqrt{\beta}})$$

We then choose β, such that $n^2 \frac{1}{\sqrt{\beta}} = n^{\frac{5}{3}} = n\beta$:

$$n^2 \frac{1}{\sqrt{\beta}} = n^{\frac{5}{3}}$$

$$\frac{1}{\sqrt{\beta}} = n^{\frac{5}{3}} \cdot \frac{1}{n^2}$$
$$\frac{1}{\sqrt{\beta}} = n^{\frac{5}{3}} \cdot n^{-2}$$
$$\frac{1}{\sqrt{\beta}} = n^{-\frac{1}{3}}$$
$$\frac{1}{\beta} = n^{-\frac{1}{3} \cdot 2}$$
$$\frac{1}{\beta} = n^{-\frac{2}{3}}$$
$$\beta = \frac{1}{n^{-\frac{2}{3}}}$$
$$\beta = n^{\frac{2}{3}}$$

So if we plug β back in we get:

$$\tilde{O}(n \cdot n^{\frac{2}{3}} + n^2 \frac{1}{\sqrt{n^{\frac{2}{3}}}})$$
$$= \tilde{O}(n^{\frac{5}{3}} + n^2 \frac{1}{n^{\frac{2}{6}}})$$
$$= \tilde{O}(n^{\frac{5}{3}} + n^{2-\frac{2}{6}})$$
$$= \tilde{O}(n^{\frac{5}{3}} + n^{\frac{5}{3}})$$
$$= \tilde{O}(n^{\frac{5}{3}})$$

b) The algorithm we design, is exactly how it's described in the exercises description. The Question is only if we really have $\tilde{O}(n^{\frac{11}{7}})$ comparisons with maximum dislocation $O(\log n)$ with probability at least $1 - \frac{1}{n}$. Lets first tackle the comparisons:

step 1: 0 comparisons

step 2: #Block $\cdot$ (#Comparisons in each Block) $= \frac{n}{\beta} \cdot \tilde{O}(\beta^{\frac{5}{3}})$ (since we use A' in each Block)

step 3: 0 comparisons

step 4: $\tilde{O}(n^2 \cdot \frac{1}{\sqrt{\beta}})$

Which totals to: $\tilde{O}(n \cdot \frac{\beta^{\frac{5}{3}}}{\beta} + n^2 \frac{1}{\sqrt{\beta}}) = \tilde{O}(n \cdot \beta^{\frac{2}{3}} + n^2 \beta^{-\frac{1}{2}})$

Again we want to choose β, such that $n\beta^{\frac{2}{3}} = n^2\beta^{-\frac{1}{2}} = n^{\frac{11}{7}}$:
We use the fact, that $\beta = n^x$ for some x, that we still need to find out:

$$n\beta^{\frac{2}{3}} = n^{\frac{11}{7}}$$
$$n \cdot n^{\frac{2x}{3}} = n^{\frac{11}{7}}$$
$$n^{\frac{3+2x}{3}} = n^{\frac{11}{7}}$$
$$\frac{3+2x}{3} = \frac{11}{7}$$
$$2x = \frac{33}{7} - \frac{21}{7} = \frac{12}{7}$$
$$x = \frac{6}{7}$$

Hence $\beta = n^x = n^{\frac{6}{7}}$
So if we plug β back in we get:

$$\begin{aligned}
&\tilde{O}(n \cdot (n^{\frac{6}{7}})^{\frac{2}{3}} + n^2 (n^{\frac{6}{7}})^{-\frac{1}{2}}) \\
&= \tilde{O}(n \cdot n^{\frac{12}{21}} + n^2 n^{-\frac{6}{14}}) \\
&= \tilde{O}(n^{1+\frac{12}{21}}) + (n^{2-\frac{6}{14}}) \\
&= \tilde{O}(n^{\frac{33}{21}}) + (n^{\frac{22}{14}}) \\
&= \tilde{O}(n^{\frac{11}{7}}) + (n^{\frac{11}{7}}) \\
&= \tilde{O}(n^{\frac{11}{7}})
\end{aligned}$$

Secondly lets tackle the probability part:
Our Algorithm uses A' on each Block. And from a)(ii), we know that A' returns with probability at least $1 - \frac{1}{n^6}$ a maximum dislocation of $O(\log n)$. Hence every P_i has a maximum dislocation of $O(\log n)$ with probability at least $1 - \frac{1}{n^6}$. So if we are lucky we are allowed to use the fact presented in the exercise description to derive that P' has a maximum dislocation of at most $c \cdot \frac{n}{\sqrt{\beta}} \log n$ with a probability of at least $1 - \frac{1}{n^8}$. So in the last step, where we call A on P', we have a probability of at least $1 - \frac{1}{n^8}$ that the result has a maximum dislocation of $O(\log n)$, if P' has a maximum dislocation of at most $c \cdot \frac{n}{\sqrt{\beta}} \log n$. So in order that our result has a maximum dislocation of $O(\log n)$ (with probability at least $1 - \frac{1}{n^8}$) we need that P' has a maximum dislocation of $c \cdot \frac{n}{\sqrt{\beta}} \log n$ (and then choose D = $c \cdot \frac{n}{\sqrt{\beta}} \log n$) and for that to happen we need that each P_i has a maximum dislocation of $O(\log n)$. Lets wrap it up:

Pr[maximum dislocation is bigger than $O(\log n)$] =
Pr[At least one P_i has a maximum dislocation greater than $O(\log n)$]
+ Pr[P' has a maximum dislocation bigger than $c \cdot \frac{n}{\sqrt{\beta}} \log n$| all P_i's have a maximum dislocation of $O(\log n)$]
+ Pr[Algo A applied on P' returns a result that has a greater maximum dislocation than $O(\log n)$| P' has a maximum dislocation of at most $c \cdot \frac{n}{\sqrt{\beta}} \log n$]
$= \frac{n}{\beta} \cdot \frac{1}{n^6} + \frac{1}{n^8} + \frac{1}{n^8}$

Since we chose $\beta = n^{\frac{6}{7}}$ in A', we have:

$$\begin{aligned}
&= \frac{n}{n^{\frac{6}{7}}} \cdot \frac{1}{n^6} + \frac{1}{n^8} + \frac{1}{n^8} \\
&= n^{\frac{1}{7}-6} + \frac{1}{n^8} + \frac{1}{n^8} \\
&\leq \frac{1}{n^5} + \frac{1}{n^8} + \frac{1}{n^8} \\
&\leq \frac{1}{n}
\end{aligned}$$

And since the error is less than $\frac{1}{n}$, the probability that we don't have an error is at least: $1 - \frac{1}{n}$

c) (i) We know:
comparisons in $A_{i+1} = \tilde{O}(n^{a_{i+1}}) = \frac{n}{\beta} \cdot \tilde{O}(\beta^{a_i}) + \tilde{O}(n^2 \frac{1}{\sqrt{\beta}}) = \tilde{O}(\frac{n}{\beta} \cdot \beta^{a_i}) + \tilde{O}(n^2 \frac{1}{\sqrt{\beta}})$

So as before we want to minimize the number of comparisons. We achieve this if $\tilde{O}(\frac{n}{\beta} \cdot \beta^{a_i}) = \tilde{O}(n^{a_{i+1}}) = \tilde{O}(n^2 \frac{1}{\sqrt{\beta}})$.

We again use the fact that $\beta = n^x$ for some x.

$$n \cdot \beta^{a_i - 1} = n^2 \beta^{-\frac{1}{2}}$$
$$n \cdot\ n^{x(a_i-1)} = n^2 n^{-\frac{x}{2}}$$
$$n \cdot\ n^{1+x(a_i-1)} = n^{-\frac{4-x}{2}}$$
$$1 + x(a_i - 1) = \frac{4-x}{2}$$
$$2 + 2xa_i - 2x = 4 - x$$
$$x = \frac{2}{2a_i - 1} = \frac{1}{a_i - 0.5}$$

Hence we get that:

$$\beta = n^x = n^{\frac{1}{a_i - 0.5}}$$

So if we plug back in our β we get:

$$\tilde{O}(n^{a_{i+1}}) = \tilde{O}(\frac{n}{n^{\frac{1}{a_i-0.5}}} \cdot (n^{\frac{1}{a_i-0.5}})^{a_i}) + \tilde{O}(n^2 \frac{1}{\sqrt{n^{\frac{1}{a_i-0.5}}}})$$

$$\tilde{O}(n^{a_{i+1}}) = \tilde{O}(n^{1-\frac{1}{a_i-0.5}} \cdot n^{\frac{a_i}{a_i-0.5}}) + \tilde{O}(n^2 \frac{1}{n^{\frac{0.5}{a_i-0.5}}})$$

$$\tilde{O}(n^{a_{i+1}}) = \tilde{O}(n^{1-\frac{1}{a_i-0.5}+\frac{a_i}{a_i-0.5}}) + \tilde{O}(n^2 \cdot n^{-\frac{0.5}{a_i-0.5}})$$

$$\tilde{O}(n^{a_{i+1}}) = \tilde{O}(n^{1+\frac{-1+a_i}{a_i-0.5}}) + \tilde{O}(n^{2-\frac{0.5}{a_i-0.5}})$$

Now we can check if $a_{i+1} = 1 + \frac{-1+a_i}{a_i-0.5} = 2 - \frac{0.5}{a_i-0.5}$:

$$1 + \frac{-1+a_i}{a_i-0.5} = \frac{a_i - 0.5 + a_i - 1}{a_i - 0.5} = \frac{2a_i - 3/2}{a_i - 1/2}$$
$$2 - \frac{0.5}{a_i - 0.5} = \frac{2(a_i - 0.5) - 0.5}{a_i - 0.5} = \frac{2a_i - 3/2}{a_i - 1/2}$$

Hence we have:

$$\tilde{O}(n^{a_{i+1}}) = \tilde{O}(n^{\frac{2a_i-3/2}{a_i-1/2}})$$

(ii) We do induction on i:

Base (i=0): $\frac{1}{2}(\frac{1}{2^{0+1}-1} + 3) = 2$ (True, since its just directly applying Algo A on S)

Hypothesis: $a_i = \frac{1}{2}(\frac{1}{2^{i+1}-1} + 3)$

Induction step: $a_{i+1} \overset{(3c(i))}{=} \frac{2a_i-3/2}{a_i-1/2} \overset{\text{I.H.}}{=} \frac{2\cdot\frac{1}{2}(\frac{1}{2^{i+1}-1}+3)-3/2}{\frac{1}{2}(\frac{1}{2^{i+1}-1}+3)-1/2} = \frac{\frac{1}{2^{i+1}-1}+3/2}{\frac{1/2}{2^{i+1}-1}+1} = \frac{\frac{1}{2^{i+1}-1}+3/2}{\frac{1}{2^{i+2}-2}+1} = \frac{\frac{1+3/2(2^{i+1}-1)}{2^{i+1}-1}}{\frac{1+2^{i+2}-2}{2^{i+2}-2}} = \frac{3/2(2^{i+1})-1/2}{(2^{i+1}-1)} \cdot \frac{2(2^{i+1}-1)}{2^{i+2}-1} = \frac{2(3/2(2^{i+1}-1/2)}{2^{i+2}-1} = \frac{3(2^{i+1})-1}{2^{i+2}-1} = \frac{3(2^{i+1})-1}{2(2^{i+1})-1} = \frac{1/2}{2(2^{i+1}-1/2)} + \frac{3(2^{i+1}-1/2)}{2(2^{i+1}-1/2)} = \frac{1/2}{2(2^{i+1}-1/2)} + \frac{3}{2} = \frac{1/2}{2^{i+2}-1} + \frac{3}{2} = \frac{1}{2}(\frac{1}{2^{(i+1)+1}-1} + 3)$

(iii) We have $\lim_{i\to\infty} a_i = \frac{3}{2}$ and its strictly monotone decreasing to $\frac{3}{2}$. Therefore, if we choose a specific h, (for example h = 17), we don't quiet have a runtime of $\tilde{O}(n^{\frac{3}{2}})$ that holds for every n. In fact for any fixed h we will never achieve this runtime, which then should hold for every n.
However we can try to give a h, depending on what n is. We want:

$$\tilde{O}(n^{\frac{1}{2}(\frac{1}{2^{h+1}-1}+3)} = \tilde{O}(n^{\frac{3}{2}})$$

$$\tilde{O}(n^{\frac{1}{2^{h+2}-2}+3/2}) = \tilde{O}(n^{\frac{3}{2}})$$

$$\tilde{O}(n^{\frac{1}{2^{h+2}-2}} \cdot n^{\frac{3}{2}}) = \tilde{O}(n^{\frac{3}{2}})$$

Hence if we achieve that $n^{\frac{1}{2^{h+2}-2}}$ is equivalent to log(?) (where ? is dependent on n. For example: ?=n) we can erase it, since it then would be a log-factor. Hence we solve for h:

$$n^{\frac{1}{2^{h+2}-2}} = \log(n)$$

$$\frac{1}{2^{h+2}-2} = \log_n(\log(n))$$

$$2^{h+2} - 2 = \frac{1}{\log_n(\log(n))}$$

$$2^{h+2} = \frac{1}{\log_n(\log(n))} + 2$$

$$h + 2 = \log_2(\frac{1}{\log_n(\log(n))} + 2)$$

$$h = \log_2(\frac{1}{\log_n(\log(n))} + 2) - 2$$

And since h is a natural number we round up:

$$h = \lceil \log_2(\frac{1}{\log_n(\log(n))} + 2) - 2 \rceil$$