MA346m Quiz 03 ANSWERS 13/11/17

(1). Consider the following

$$\text{Subject to } 1 \le x_1 < x_2 < \ldots < x_r \le n$$
$$\text{Minimise } f(x_1, \ldots, x_r) =$$
$$\log_2 x_1 + \log_2 x_2 + \ldots \log_2 x_r - \log_2(x_2 - x_1) - \ldots - \log_2(x_r - x_{r-1}).$$

by equating the partial derivatives $\partial f / \partial x_j$ to zero, for $2 \le j \le r - 2$.

Since the natural log ln and $\log_2$ are proportional, there is a small simplification in replacing $\log_2$ by ln in the calculations.

Deduce that to achieve a minimum, the ratios x_j / x_{j-1} must be constant, $2 \le j \le r - 1$, so most of the sequence $x_1, \ldots, x_r$ is in geometric progression.

(I'm not absolutely sure of how x_1, x_{r-1}, and x_r, are treated, but there should be enough here to handle the next question.)

Answer. For $2 \le j \le r - 1$,

$$\frac{\partial f}{\partial x_j} = \frac{1}{x_j} - \frac{1}{x_j - x_{j-1}} + \frac{1}{x_{j+1} - x_j} = 0$$
$$(x_{j+1} - x_j)(x_j - x_{j-1}) - x_j(x_{j+1} - x_j) + x_j(x_j - x_{j-1}) = 0$$
$$-x_{j-1}x_{j+1} + x_j^2 = 0$$
$$\frac{x_{j+1}}{x_j} = \frac{x_j}{x_{j-1}}. \quad \blacksquare$$

(2).

Continuing the above. Suppose $r - 2 \ge \log_2 n$, and define s as the ratio of the above geometric sequence, so

$$s^{r-2} x_1 = x_{r-1}$$

(typo corrected?)

So $x_1 s^{r-2} = x_{r-1}$. From the above result, given: $r - 2 \ge \log_2 n$,

$$s^{r-2} \le x_1 s^{r-2} \le x_{r-1} \le n$$
$$(r-2) \log_2 s \le \log_2 n \le r - 2$$
$$\log_2 s \le 1$$
$$s \le 2$$

Assume $r - 2 \ge \log_2 n$. Deduce that $f(x_1, \ldots, x_r) \ge r - 2$ (give or take a $\log_2 n$, say).

$$\log x_1 + \sum_{j=2}^{r-1} \log x_j - \log(x_j - x_{j-1}) + \log x_r - \log(x_r - x_{r-1}) \ge$$
$$\log x_1 + (r-2) \log \left(\frac{1}{1 - \frac{1}{s}} \right) + \log x_r - \log(x_r - x_{r-1}) \ge r - 2.$$

(Logs to the base 2.)

(3). The above can be used to analyse the cost of the Union-Find strategy applied *with* path-compression but *without* size-balancing. Use a potential function like that used in splay trees: $\Phi = \sum_v \log_2 \text{size}(v)$, where size$(v)$, the size of v, is the number of descendants v currently possesses.

Answer. Cost of unions is $O(n)$. Unions can increase Φ, but by at most $\log_2 n$ each time. Finds cannot increase Φ, and a find of length $r =\geq \log_2 n$ reduces the potential by at least $r-2$. A find of length $\leq \log_2 n$ costs $O(\log n)$. In any event, every find has $O(\log n)$ amortised cost. The total potential gain is at most $n \log n$, which accounts for most of the cost of a long find, involving $r \geq \log_2 n$ nodes. The overall cost of $n-1$ unions and m finds is at most

$$n \log_2 n + 2m + m \log_2 n + n = O((m+n)\log n). \quad \blacksquare$$

(4). Under linear collision resolution, the expected cost S_n of successful search in a hash table of size m with n slots filled is

$$S_n = \frac{1}{n}\sum_{k=0}^{n-1} U_k = \frac{1}{2} + \frac{m-1}{2(m-n)}$$

where U_n is the expected cost of an unsuccessful search. Deduce that

$$U_n = \frac{1}{2} + \frac{m(m-1)}{2(m-n-1)(m-n)}.$$

Answer.

$$U_n = (n+1)S_{n+1} - nS_n =$$

$$(n+1)\left(\frac{1}{2} + \frac{1}{2}\frac{m-1}{m-n-1}\right) - n\left(\frac{1}{2} + \frac{1}{2}\frac{m-1}{m-n}\right) =$$

$$\frac{1}{2} + \frac{m-1}{2}\left(\frac{n+1}{m-n-1} - \frac{n}{m-n}\right) =$$

$$\frac{1}{2} + \left(\frac{m-1}{2}\right)\left(\frac{mn+m-n^2-n-mn+n^2+n}{(m-n-1)(m-n)}\right) = \frac{1}{2} + \frac{m(m-1)}{2(m-n-1)(m-n)}.$$

(5). What is the cost of Union-Find using size balancing but *without* path compression?
Answer. $O(m\log n + n)$.