CS 170

2. Asymptotic Properties

(a) Induction on n:
when $n = 1$, it is trivial that $1 = \Theta(1)$
if the property holds for $n - 1$, we get

$$\begin{aligned}\sum_{i=1}^{n} i^c &= \sum_{i=1}^{n-1} i^c + n^c \\ &= \Theta((n-1)^{c+1}) + n^c \\ &= \Theta(n^{c+1})\end{aligned}$$

So, the property holds for all $n > 0$.

(b) easy part, refer to the solution for help

3. Median of Medians

(a) the worst-case runtime is $\mathcal{O}(n^2)$
One bad pivot choice may be $1, 2, 3, \ldots, \lfloor \frac{n}{2} \rfloor$.

(b) proof :
Since p is the median of the medians, and there are $\frac{n}{5}$ medians, so there are $\frac{n}{10}$ medians fewer than p. For each median, in its own 5-element sequence, there are three elements fewer than it, So there are at least $\frac{n}{10} * 3 = \frac{3n}{10}$ elements fewer than p.
Similarly, we can prove that there are at least $\frac{3n}{10}$ elements greater than p.

(c) $\mathrm{T}(n) \leq \mathrm{T}(\frac{7n}{10}) + \mathrm{T}(\frac{n}{5}) + d * n$
using induction can easily prove that there exists c to satisfy $\mathrm{T}(n) \leq c * n$

4. Werewolves

the same as the classic "chip testing" problem.

5. Hadamard matrices

(a) $H_0 = \begin{bmatrix} 1 \end{bmatrix}$

$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

$H_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$

(b) $H_2 \cdot v = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 4 \end{pmatrix}$

(c) $u = H_2 \cdot v$

(d) $H_k \cdot v = \begin{bmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & -H_{k-1} \end{bmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} H_{k-1}(v_1 + v_2) \\ H_{k-1}(v_1 - v_2) \end{pmatrix}$

(e) let $\mathrm{T}(k)$ to be the runtime to compute $H_k \cdot v$
from the equation from the above part, we can divide H_k into two matrix vector multiplication, each costs time $\mathrm{T}(k-1)$. Note that we also need one vector addition and subtraction, which costs time n in total.
So $\mathrm{T}(k) = 2\,\mathrm{T}(k-1) + \mathcal{O}(n)$, using master theorem we get $\mathrm{T}(n) = \mathcal{O}(nlog(n))$.