Problem 1.

a) **False.** Let $h(n) = n^3 + 2n^2$ and let $f(n) = 0.1n^{3.1} + n$. We know that $h(n) \in \omega(f(n))$ if $\lim_{n\to\infty} \frac{h(n)}{f(n)} = \infty$. Evaluating the limit shows

$$\begin{aligned}
\lim_{n\to\infty} \frac{h(n)}{f(n)} &= \lim_{n\to\infty} \frac{n^3 + 2n^2}{0.1n^{3.1} + n} \\
&= \lim_{n\to\infty} \frac{\frac{n^3}{n^{3.1}} + \frac{2n^2}{n^{3.1}}}{\frac{0.1n^{3.1}}{n^{3.1}} + \frac{n}{n^{3.1}}} \\
&= \lim_{n\to\infty} \frac{0+0}{0.1+0} \\
&= 0.
\end{aligned}$$

Hence $h(n) \notin \omega(f(n))$.

b) **False.** Let $h(n) = n\log^5(n)$ and let $f(n) = n^{\frac{5}{4}}$. There are no constants c_0, c_1 such that $c_0 f(n) \le h(n) \le c_1 f(n)$. We would need to find c_0, c_1 such that

$$c_0 n^{\frac{5}{4}} \le n\log^5(n) \le c_1 n^{\frac{5}{4}}$$

which is equivalent to

$$c_0 n^{\frac{1}{4}} \le \log^5(n) \le c_1 n^{\frac{1}{4}}.$$

This is impossible because there is not some c_0, c_1 such that the equality holds for all $n \ge n_0$ for some $n_0 \in \mathbb{N}$ since $n^{\frac{1}{4}}$ grows at a different rate than $\log^5(n)$. Hence $h(n) \notin \Theta(f(n))$.

c) **False.** Let $h(n) = 2^{n\log(n)}$ and let $f(n) = 4^n$. Notice that $h(n) = 2^{\log(n^n)} = (n^n)^{\log(2)} = n^n$. It is clear there are no constants c_0, c_1 such that $c_0 f(n) \le h(n) \le c_1 f(n)$ for all $n \ge n_0$ for some $n_0 \in \mathbb{N}$ since n^n grows at a faster rate than 4^n. Hence $h(n) \notin \Theta(f(n))$.

d) **True.** We can apply the Sterling approximation $n! \approx \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$. The dominating factor of the right hand side is $\left(\frac{n}{e}\right)^n$ which grows faster than a^{n+1} for any $a > 1$ given sufficiently large n. Hence $a^{n+1} \in o(n!)$.

e) **True.** Let $h(n) = \sqrt{\log(n)}$ and let $f(n) = \log(\sqrt{n})$. Notice $f(n) = \frac{1}{2}\log(n)$. We have that

$$\begin{aligned}
\lim_{n\to\infty} \frac{h(n)}{f(n)} &= \lim_{n\to\infty} \frac{\sqrt{\log(n)}}{\frac{1}{2}\log(n)} \\
&= \lim_{n\to\infty} \frac{2}{\sqrt{\log(n)}} \\
&= 0.
\end{aligned}$$

Hence $h(n) \in o(f(n))$.

Problem 2.

1. $1/\log(n)$
2. $2^{2+1/n}$
3. $\sqrt{\log(\log(n))}$
4. $\log(n)/\log(\log(n))$
5. $\log(n)$
6. $3^{\sqrt{\log(n)}}$
7. $\{e^{\ln(n)}, 5(n+1/n)\}$
8. $\log(n!)$
9. $n\log^3(n)$
10. $4n^{\frac{3}{2}}$
11. $4^{\log(n)}$
12. $n^2\log(n)$
13. n^3
14. n^{100}
15. $3^{\sqrt{n}}$
16. 4^n
17. $n!$
18. $2^{n^2\log(n)}$
19. 2^{2^n}

Problem 3.

a) **False.** Let $f(n) = n^2$ and let $g(n) = 2n^2$. We can show that $f(n) \in O(g(n))$ and $g(n) \in O(f(n))$:

- $f(n) \in O(g(n))$: let us show there exists $c > 0$, $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$ we have $f(n) \leq cg(n)$. Let $c = \frac{1}{2}$. Then we have

  $$n^2 \leq c2n^2 = \frac{1}{2}2n^2 = n^2$$

  which is true for any $n_0 \in \mathbb{N}$. Hence $f(n) \in O(g(n))$.

- $g(n) \in O(f(n))$: let us show there exists $c > 0$, $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$ we have $g(n) \leq cf(n)$. Let $c = 2$. Then we have

  $$2n^2 \leq cn^2 = 2n^2$$

  which is true for any $n_0 \in \mathbb{N}$. Hence $g(n) \in O(n(n))$.

This shows it is possible for $f(n) \in O(g(n))$ and $g(n) \in O(n(n))$.

b) **True.** $f(n)+g(n) \in \Theta(f(n))$ if there exists $c_0 > 0$, $c_1 > 0$, $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$ we have $c_0 f(n) \leq f(n) + g(n) \leq c_1 f(n)$. We know for a fact that $f(n) + g(n) \leq 2f(n)$. Hence, let us choose $c_0 = 1$ and $c_1 = 2$. Then we have

$$f(n) \leq f(n) + g(n) \leq 2f(n).$$

This is clearly true as $g(n) \geq 0$ for all n, thus showing $f(n) + g(n) \in \Theta(f(n))$.

c) **False.** We prove that the statement is false through a counterexample. Let $f(n) = 4n$ and let $g(n) = n$. It is trivial that $f(n) \in \Theta(n)$ and $g(n) \in \Theta(n)$. In order for $2^{f(n)} \in \Theta(2^{g(n)})$, there must exist $c_0 > 0$, $c_1 > 0$, $n_0 \in \mathbb{N}$ such that

$$c_0 2^{g(n)} \leq 2^{f(n)} \leq c_1 2^{g(n)}$$

for all $n \geq n_0$. When we substitute in $f(n)$ and $g(n)$, we get

$$c_0 2^n \leq 2^{4n} \leq c_1 2^n.$$

Taking the logarithm of each expression yields

$$\log(c_0 2^n) \leq \log(2^{4n}) \leq \log(c_1 2^n)$$

which can be simplified to

$$\log(c_0) + n \leq 4n \leq \log(c_1) + n.$$

Subtracting n from each expression yields

$$\log(c_0) \leq 3n \leq \log(c_1).$$

There is no selection of $c_0 > 0$, $c_1 > 0$, $n_0 \in \mathbb{N}$ such that the inequality holds for all $n \geq n_0$ because $3n \to \infty$ as $n \to \infty$.

d) **False.** Let $f(n) = \frac{1}{n}$. Then, in order for $f(n) \in \Theta((f(n))^2)$, there must exist $c_0 > 0$, $c_1 > 0$, $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$ we have

$$\frac{c_0}{n^2} \leq \frac{1}{n} \leq \frac{c_1}{n^2}.$$

This is equivalent to

$$\frac{c_0}{n} \leq 1 \leq \frac{c_1}{n}.$$

However, there is no value of c_0 or c_1 that can satisfy the inequality for all $n \geq n_0$ where $n_0 \in \mathbb{N}$ because $\frac{c_1}{n} \to 0$ as $n \to \infty$.

Problem 4.

a) As described in the algorithm, we require $\frac{n}{k} + k \in O(\sqrt{n})$. We note that the only choice of k that satisfies the requirement is $k = \sqrt{n}$ as this results in the algorithm requiring $\frac{n}{\sqrt{n}} + \sqrt{n} = \sqrt{n} + \sqrt{n} = 2\sqrt{n}$ touches which is in $O(\sqrt{n})$.

Let $k = \sqrt{n}$. We use the first wand on boxes $1, k, 2k, 3k, \ldots$ sequentially until one of two scenarios occur:

- $ik \leq n$ and the wand disappears. This indicates the first empty box is among $(i-1)k + 1, \ldots, ik$. We then use the second wand on boxes $(i-1)k+1, \ldots, ik$ sequentially until the wand disappears. The point at which the second wand disappears is where the empty boxes start.
- $ik > n$. This indicates the first empty box is among $(i-1)k+1, \ldots, n$. We then use the second wand on boxes $(i-1)k+1, \ldots, n$ sequentially until the wand disappears. The point at which the second wand disappears is where the empty boxes start.

For either scenario, the maximum number of touches of the first wand is $\frac{n}{k} = \frac{n}{\sqrt{n}} = \sqrt{n}$ and the maximum number of touches of the second wand is $k = \sqrt{n}$. As a result, the total number of touches for any scenario will be at most $\sqrt{n} + \sqrt{n} = 2\sqrt{n}$. We can show that this uses no more than $O(\sqrt{n})$ touches by showing that there exists $c > 0$, $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$ we have $2\sqrt{n} \leq c\sqrt{n}$. This is trivial: we can select $c = 2$ to satisfy the inequality for any $n_0 \in \mathbb{N}$.

b) **Wand 1:** Let $k = n^{2/3}$. We use the first wand on boxes $1, k, 2k, 3k, \ldots$ sequentially until a wand disappears. Let i be defined such that ik is the box at which the wand disappears. This indicates the first empty box is among subsequence $(i-1)k+1, \ldots, ik$. It takes at most $\frac{n}{k}$ touches with the wand to find this subsequence and the subsequence contains at most k elements.

Wand 2: Now let $q = \sqrt{k}$. We repeat the previous process with the second wand on boxes

$$[(i-1)k+1]+1, [(i-1)k+1]+q, [(i-1)k+1]+2q, \ldots$$

sequentially until the second wand disappears. Let j be defined such that $[(i-1)k+1]+jq$ is the box at which the wand disappears. This indicates the first empty box is among the subsequence $[(i-1)k+1]+(j-1)q+1, \ldots, [(i-1)k+1]+jq$. It takes at most $\frac{k}{q}$ touches with the wand to find this subsequence and the subsequence contains at most q elements.

Wand 3: Finally, we use the third wand on the subsequence $[(i-1)k+1]+(j-1)q+1, \ldots, [(i-1)k+1]+jq$ sequentially until the wand disappears. The point at which the wand disappears is where the empty boxes start. It takes at most q touches with the wand to find the first empty box.

Analysis: The total touches required is given by

$$\begin{aligned}\frac{n}{k}+\frac{k}{q}+q &= \frac{n}{n^{2/3}}+\frac{n^{2/3}}{\sqrt{k}}+\sqrt{k}\\ &= \frac{n}{n^{2/3}}+\frac{n^{2/3}}{\sqrt{n^{2/3}}}+\sqrt{n^{2/3}}\\ &= n^{1/3}+n^{1/3}+n^{1/3}\\ &= 3n^{1/3}\\ &= 3\sqrt[3]{n}.\end{aligned}$$

Hence, the upper bound in running time is $O(\sqrt[3]{n})$. We can show this by showing that there exists $c > 0$, $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$ we have $3\sqrt[3]{n} \leq c\sqrt[3]{n}$. This is trivial: we can select $c = 3$ to satisfy the inequality for any $n_0 \in \mathbb{N}$.

c) Since all empty boxes are adjacent to each other, we know that if box i is the first empty box, either $i = 1$ or box $i - 1$ contains a pearl. Let us define $f(n)$ such that $f(n)$ is true if box n is empty and either $n = 1$ or box $n - 1$ contains a pearl (i.e. $f(n)$ is true if box n is the first empty box). Let $l = 1$ and let $r = n$. While $l \leq r$, we repeat:

1) Let $m = \text{floor}\left(\frac{l}{2}+\frac{r}{2}\right)$.
2) If $l = r$ or $f(m)$, then we have found the first empty box. Terminate and return m.
3) If box m is empty, then we know the first empty box is left of m meaning the first empty box is in subsequence $l, l+1, \ldots, m-2, m-1$. Set $r = m - 1$.
4) Otherwise, box m contains a pearl. Then we know the first empty box is right of m meaning the first empty box is in subsequence $m+1, m+2, \ldots, r-1, r$. Set $l = m + 1$.

The psuedocode is shown below.

```
procedure FindFirstEmpty($A, n$)
    $l \leftarrow 1$
    $r \leftarrow n$
    while ($l \leq r$) do
        $m \leftarrow$ floor($l/2 + r/2$)
        if ($l = r$) or (Empty($A[m]$) and ($m = 1$ or NotEmpty($A[m-1]$)) then
            return $m$
        end if
        if (Empty($A[m]$)) then
            $r \leftarrow m - 1$
        else
            $l \leftarrow m + 1$
        end if
    end while
```

In the worst case, the number of magic wands required is in $O(\log(n))$. This occurs if all n boxes are empty as the algorithm will check floor($\log(n)$) boxes (all of which consume a magic wand to check) before reaching the first box (at which point $l = r = 1$ and checking the box with a wand is not needed).

In the best case, the number of magic wands required is in $O(1)$. This occurs if the first box we check is the first empty box: checking the empty box itself requires the destruction of one magic wand while checking the box to the left does not destroy the magic wand since it is not empty.