In [ ]:

```
import math
import numpy as np
import matplotlib.pyplot as plt
```

In [ ]:

```
def plot_theta(f, g, cp, cpp, n_0, min_x, max_x):
    xs = np.arange(min_x,max_x,1, dtype='int64')
    fn_ys = f(xs)
    gn_ysp = cp*g(xs)
    gn_yspp = cpp*g(xs)
    plt.axvline(x = n_0, color = 'k', linestyle='--')
    plt.scatter(x=xs, y=fn_ys, marker='x')
    plt.scatter(x=xs, y=gn_ysp, marker='.')
    plt.scatter(x=xs, y=gn_yspp, marker='.')
    plt.xlabel('n')
    plt.ylabel('$\lambda(n)$')
    plt.legend(['n_0 = '+str(n_0),'f(n)', 'c\'g(n)', 'c\'\'g(n)'])


def plot_oh_incr(f, g, c, n_0, min_x, max_x, incr):
    xs = np.arange(min_x,max_x,incr, dtype='int64')
    fn_ys = f(xs)
    gn_ys = c*g(xs)
    plt.axvline(x = n_0, color = 'k', linestyle='--')
    plt.scatter(x=xs, y=fn_ys, marker='x')
    plt.scatter(x=xs, y=gn_ys, marker='.')
    plt.xlabel('n')
    plt.ylabel('$\lambda(n)$')
    plt.legend(['n_0 = '+str(n_0),'f(n)', 'cg(n)'])

def plot_oh(f, g, c, n_0, min_x, max_x):
    plot_oh_incr(f, g, c, n_0, min_x, max_x, 1)
```

Big-Oh Definition

Given positive functions $f(n)$ and $g(n)$, we can say that $f(n)$ is $O(g(n))$ if and only if there exists positive constants c and n_0 such that:

$$f(n) \leq c \cdot g(n), \forall n \geq n_0$$

Big-Oh Using Rules

Drop smaller terms rule:

If $f(n) = (1 + h(n))$ and $h(n) \rightarrow 0$ as $n \rightarrow \infty$, then $f(n)$ is $O(1)$

Q1. Prove that $n^3 + 2n^2$ is $O(n^3)$ using the multiplication and drop smaller terms rules

Q2. Prove that $n^3 + 2n^2 log(n)$ is $O(n^3)$ using the multiplication and drop smaller terms rules

Big-Omega and Big-Theta Definitions

Big-Omega: Given positive functions $f(n)$ and $g(n)$, we can say that $f(n)$ is $\Omega(g(n))$ if and only if there exists strictly positive constants c and n_0 such that:

$$f(n) \geq c \cdot g(n), \forall n \geq n_0$$

Ω expresses that a function $f(n)$ grows at least as fast as $g(n)$.

Big-Theta: Given positive functions $f(n)$ and $g(n)$, we can say that $f(n)$ is $\Theta(g(n))$ if and only if there exists positive constants c', c'' and n_0 such that:

$$f(n) \leq c' \cdot g(n),\ f(n) \geq c'' \cdot g(n),\ \forall n \geq n_0$$

Θ expresses that a function $f(n)$ grows exactly as fast as $g(n)$.

Q3. Prove that $2n + 1$ is $\Omega(3n)$ and hence $2n + 1$ is $\Theta(3n)$

Plot Ω

In [ ]:
```
# c = ???
# n_0 = ???
# f = lambda n: ???
# g = lambda n: ???

plot_oh(f, g, c, n_0, 0, 50)
```

Plot Θ

In [ ]:
```
# cp = ???
# cpp = ???
# n_0 = ???
# f = lambda n: ???
# g = lambda n: ???

plot_theta(f, g, cp, cpp, n_0, 0, 50)
```

Big-Omega/Big-Theta Questions: Basics

Q4. Prove that 5 is $\Omega(1)$, and hence 5 is $\Theta(1)$

Q5. Prove that 4 is $\Omega(2)$, and hence 4 is $\Theta(2)$

Q6. Prove that $2n + 1$ is $\Omega(n)$, and hence $2n + 1$ is $\Theta(n)$

Big-Omega/Big-Theta Questions: Medium Difficulty

Q7. Prove that n^2 is $\Omega(2n^2)$

Q8. Prove that $n^2 - 3$ is $\Omega(n^2)$

Q9. Prove that $n^2 - 5n$ is $\Omega(n^2)$, and hence is $\Theta(n^2)$

Q10. Prove that $n^2 + 1$ is $\Omega(n^2)$

Little-Oh Definition

Given positive functions $f(n)$ and $g(n)$, we can say that $f(n)$ is $o(g(n))$ if **for all positive real constants** $c > 0$ there exists n_0 such that:

$$f(n) < c \cdot g(n), \forall n \geq n_0$$

Important:

- $c \in \R_*^+$
- $\R_*^+ = \{x \in \R | x > 0\}$

Q11. Prove or disprove that 5 is $o(1)$

Q12. Prove or disprove that 5 is $o(n)$

Little-Oh Questions

Q13. Prove or disprove that n is $o(n^2)$

Q14. Prove or disprove that 1 is $o(log\ n)$

Q15. Prove or disprove that $log\ n$ is $o(1)$

Additional Practice Questions (more challenging)

Q16. Prove or disprove that 1 is $\Omega(n)$

Q17. Prove or disprove that n is $\Omega(1)$

Q18. Prove or disprove that n^2 is $\Omega(n)$

Q19. Prove or disprove that n is $o(n\ log\ n)$

Q20. Given that $f(n) = n^2$ if n is even, and $f(n) = n$ if n is odd. From the definitions find the O and Ω behaviours of $f(n)$.

Warning: be careful to find a single c that works for all n, not separate c for even and odd n

Q21. Prove or disprove that $n\ log\ n$ is $o(n^2)$