$$\Omega\big(g(n)\big) = \exists n_0, c\,.\forall n > n_0\,. \quad 0 \le cg(n) \le f(n)$$
$$O\big(g(n)\big) = \exists n_0, c\,.\forall n > n_0\,. \quad 0 \le f(n) \le cg(n)$$
$$n^{2+\sin\left(n\frac{\pi}{2}\right)} = \Omega(n^a) \implies \exists n_0, c\,.\forall n > n_0\,. \quad 0 \le cn^a \le n^{2+\sin\left(n\frac{\pi}{2}\right)}$$
$$-1 \le \sin(n\frac{\pi}{2}) \le 1 \implies 1 \le 2+\sin(n\frac{\pi}{2}) \le 3$$
$$\implies n^1 \le 2+\sin(n\frac{\pi}{2}) \le n^3 \Rightarrow \max(a) = 1, \min(b) = 3$$

فرض کنیم a>1

اگر $n = 4k + 3$ باشد به ازای k های صحیح $f(n) = n$ می‌شود و انوقت نمی‌توان گفت که $f(n) = \Omega(n^a)$ چرا که به ازای هر ضریب c و n0 وجود دارد حالتی که $f(n) < cg(n)$

برای عدد b باید اثبات کنیم از این کمتر نمی‌تواند باشد یعنی ۱ و ۲

اگر $n = 4k+1$ به ازای k های صحیح $f(n) = n^3$ می‌شود و برای $b < 3$، $f(n)=O(n^b)$ رابطه به هیچ عنوان برقرار نیست چرا که به ازای هر ضریب c و n0 وجود دارد حالتی که $cg(n) < f(n)$

۲) عبارت زیر را ثابت کنید. (تابع F(n) همان فیبوناچی n ام است؛ یعنی $F(n) = F(n-1) + F(n-2)$ و $F(1)=F(2)=1$) *(۱ نمره)*

$$F(n)=O((\sqrt{3})^n)$$

$$F(n) = F(n-1) + F(n-2) \quad \Rightarrow r^2 = r + 1 \Rightarrow r = \frac{1 \pm \sqrt{5}}{2}$$

$$F(n) = c_1\left(\frac{1+\sqrt{5}}{2}\right)^n + c_2\left(\frac{1-\sqrt{5}}{2}\right)^n$$

$$\begin{cases} F(1) = 1: \ c_1\left(\frac{1+\sqrt{5}}{2}\right) + c_2\left(\frac{1-\sqrt{5}}{2}\right) = 1 \\ F(2) = 1: \ c_1\left(\frac{3+\sqrt{5}}{2}\right) + c_2\left(\frac{3-\sqrt{5}}{2}\right) = 1 \end{cases} \Rightarrow c_1 = \frac{1}{\sqrt{5}}, \qquad c_2.-\frac{1}{\sqrt{5}}$$

$$F(n) = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$$

$$\lim_{n\to\infty} \frac{\frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n}{\sqrt{3}^n} = \lim_{n\to\infty} \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2\sqrt{3}}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2\sqrt{3}}\right)^n$$

$$\frac{1+\sqrt{5}}{2\sqrt{3}} \cong \frac{3.2}{3.4} < 1, \qquad \frac{1-\sqrt{5}}{2\sqrt{3}} \cong \frac{1.2}{3.4} < 1 \Rightarrow \lim_{n\to\infty}\left(\frac{1 \pm \sqrt{5}}{2\sqrt{3}}\right)^n = 0$$

$$\Rightarrow \lim_{n\to\infty} \frac{F(n)}{\sqrt{3}^n} = 0 \Rightarrow F(n) = O(\sqrt{3}^n)$$

است.) (۲ نمره)

$$a^n = o(n!) \Rightarrow \ \exists c, n_0 \in N \,.\, \forall n > \ n_0 \,. \quad 0 \leq \ a^n < c.n!$$

راه حل اول: تقریب استرلینگ

$$\lim_{n \to \infty} \frac{a^n}{n!} \cong \ \lim_{n \to \infty} \frac{a^n}{\sqrt{2\pi n}\ \left(\frac{n}{e}\right)^n} = \ \lim_{n \to \infty} \frac{(ae)^n}{\sqrt{2\pi n}\ n^n} = \ \lim_{n \to \infty} \frac{1}{\sqrt{2\pi n}} \cdot \left(\frac{ae}{n}\right)^n$$

$$\lim_{n \to \infty} \frac{ae}{n} = 0 \ \Longrightarrow \lim_{n \to \infty} \left(\frac{ae}{n}\right)^n = 0 \ \Longrightarrow \lim_{n \to \infty} \ \frac{1}{\sqrt{2\pi n}} . \left(\frac{ae}{n}\right)^n = 0$$

$$\Longrightarrow a^n = o(n!)$$

راه حل دوم:

$$a^n < c.n!$$

$$\log(a^n) < \log(c.n!) \ \Longrightarrow n \log a < \log(c) + \log(n) + \cdots + \log(1)$$

با استقرا عبارت بالا را $c = 1$ اثبات می‌کنیم

پایه استقرا : $\ n = a \ \Rightarrow \log(a) < \log(1) + \log(a) + \log(a-1) + \cdots + \log(1)$

فرض استقرا: $\ n \log a < \log(1) + \log(n) + \cdots + \log(2) + \log(1)$

حکم استقرا: $(n+1)\log(a) < \ \log(1) + \log(n+1) + \log(n-1) + \log(1)$

$$\begin{cases} n \log a < \log(1) + \log(n) + \cdots + \log(2) + \log(1) \\ \qquad\qquad\qquad \log(a) < \log(n+1) \end{cases} \overset{+}{\Rightarrow}$$

$$(n+1)\log(a) < \ \log(1) + \log(n+1) + \log(n-1) + \log(1)$$

حکم استقرا ثابت شد در نتیجه:

$$a^n = o(n!)$$

۴) ثابت کنید: $g(n) = \Omega(f(n)) \Rightarrow g(n) = \Omega(O(f(n)))$ *(۱ نمره)*

طبق تعریف Ω و O میدانیم:

$g(n) = \Omega(f(n)) \Rightarrow \exists c, n0:\ \forall n \geq n0.\ 0 \leq c.g(n) \leq f(n)$ (1)

$h(n) = O(f(n)) \Rightarrow \exists d, n'0:\ \forall n \geq n'0.\ 0 \leq f(n) \leq d.h(n)$ (2)

بنابراین:

$(1), (2) \Rightarrow 0 \leq c.g(n) \leq f(n) \leq d.h(n) \Rightarrow 0 \leq c.g(n) \leq d.h(n) \Rightarrow$

$\exists c, d, n0, n'0:\ \forall n \geq \max(n0, n'0).\ 0 \leq (\frac{c}{d}).g(n) \leq h(n)$ (3)

$(3) \Rightarrow g(n) = \Omega(h(n)) = \Omega(O(f(n)))$

$\Rightarrow g(n) = \Omega(O(f(n)))$

۵) به ازای دو عدد حقیقی a و b به طوری که $a > b > 1$ ثابت کنید: $(n > 1)$ *(۰.۵ نمره)*

$$n * \log(n, b) = \theta(n * \log(n, a))$$

جواب:

۵) باید ثابت کنیم: $\lim\limits_{n\to\infty} \frac{n\log_a^n}{n\log_b^n} = c$

$$\Rightarrow \lim_{n\to\infty} \frac{n\log_a^n}{n\log_b^n} = \lim \frac{\frac{\log^n}{\log a}}{\frac{\log^n}{\log b}} = \frac{\log b}{\log a} = c \quad \checkmark$$

۱. T(n) = T(n/۲) + ۱, T(۱) = ۱

$$T(n) = cn + \frac{cn}{2} + \frac{cn}{4} + \frac{cn}{8} + \cdots + \left(\frac{cn}{2}\right)^{\log_2 n-1} \leq T\left(\frac{n}{2}\right) + cn \leq d\left(\frac{n}{2}\right)\log_2\frac{n}{2} + cn = \boldsymbol{\Theta(\log_2 n)}$$

۲. T(n) = ۳T(n/۲) + ۱, T(۱) = ۱

$$T(n) = cn + 3\frac{cn}{2} + 9\frac{cn}{4} + 27\frac{cn}{8} + \cdots + \left(3\frac{cn}{2}\right)^{\log_2 n-1} + \Theta(n^{\log_2 3}) \leq 3T\left(\frac{n}{2}\right) + cn \leq d.3.\left(\frac{n}{2}\right)\log_2\frac{n}{2} + cn$$
$$= \boldsymbol{\Theta(n^{\log_2 3})}$$

۳. T(n) = T(n-۱) + n, T(۱) = ۱

$$\mathrm{T(n) = T(n-1) + n = (T(n-2) + n) + n = \cdots = T(n-k) + nk} \rightarrow \frac{\mathrm{n(n+1)}}{2} \rightarrow \boldsymbol{\theta(\mathrm{n}^2)}$$

```
COUNT_INVERSIONS(A)
    inversion_count = 0
    for i=1 to A.size
        for j=1 to i
            if  A[i] < A[j]
                inversion_count = inversion_count + 1
    return inversion_count
```

مرتبه زمانی: $(\mathrm{T(n)} \in \mathrm{O}(n^2))$

```
print_divisibility (arr[])
    for i = 1 to 1000
        flag = true
        for j = 0 to arr.length-1
            if i % arr[j] != 0
                flag = false
                break
        if flag
            print(i)
```

دستورات داخل حلقه اول (با متغییر i)، 1000 بار تکرار می‌شوند دستورات داخلی حلقه دوم و داخلی به اندازه طول آرایه در ۱۰۰۰ تکرار می‌شوند یعنی n بار در نتیجه:

```
T(n) = a*1000*n + b*1000 → T(n) = Θ(n)
```

۹) کامپیوتری داریم که در هر ثانیه 10^{12} عملیات انجام می‌دهد. دو الگوریتم داریم که به ازای ورودی n، به ترتیب n^2 و 2^n مرحله اجرایشان طول می‌کشد. به ازای هر الگوریتم مشخص کنید که اگر بخواهیم اجرا ۱ ثانیه و یا ۱۰۲۴ ثانیه طول بکشد، باید چه مقداری به ورودی آن بدهیم. *(برای راحتی بیشتر می‌توانید فرض کنید که 2^{40} با 10^{12} تقریباً برابر است.) (هر مورد ۰.۲۵، در مجموع ۱ نمره)*

n^2	$1s$	$n^2 \times \frac{1}{10^{12}} = 1s \Rightarrow n^2 = 10^{12} = (10^6)^2 \Rightarrow \boldsymbol{n = 10^6}$
	$2^{10}s$	$n^2 \times \frac{1}{10^{12}} = 2^{10}s \Rightarrow n^2 = 2^{10} \times 10^{12} \cong 2^{10+40} = (2^{25})^2 \Rightarrow \boldsymbol{n = 2^{25}}$
2^n	$1s$	$2^n \times \frac{1}{10^{12}} = 1s \Rightarrow 2^n = 10^{12} \cong 2^{40} \Rightarrow \boldsymbol{n = 40}$
	$2^{10}s$	$2^n \times \frac{1}{10^{12}} = 2^{10}s \Rightarrow 2^n = 2^{10} \times 10^{12} \cong 2^{10+40} = 2^{50} \Rightarrow \boldsymbol{n = 50}$