أوو، كم هي جميلة هذه الدالة
Oh, What a Beautiful Function

أوجدنا من فترة طويلة صيغة إيجاد مجموع أول n عدد صحيح:

$$1+2+3+\cdots+(n-1)+n=\frac{n(n+1)}{2}=\frac{1}{2}n^2+\frac{1}{2}n$$

هذه صيغة دقيقة تماماً وجميلة جدًّا، لكن قد تكون هناك حالات نفضل فيها صيغة أقل تعقيداً، حتى لو فقدنا بعض الدقة. لذلك يمكننا أن نقول إن $1+2+\cdots+n$ تقريباً يساوي $\frac{1}{2}n^2$، حيث إنه عندما يكون n كبيراً فإن الحد $\frac{1}{2}n^2$ أكبر بكثير من الحد $\frac{1}{2}n$.

نفس الشيء، هناك صيغة دقيقة لمجموع مربعات أول n حد[1]،

$$1^2+2^2+\cdots+(n-1)^2+n^2=\frac{n(n+1)(2n+1)}{6}$$

(١) لقد أثبت ذلك سابقاً إذا كنت قد قمت بحل التمرين 7.3. إذا لم تقم بحله فستجد البرهان باستخدام طريقة مختلفة وذلك في الفصل الثاني والأربعون.

وعليه ؛ فإن :

$$1^2+2^2+\cdots+(n-1)^2+n^2=\frac{1}{3}n^3+\frac{1}{2}n^2+\frac{1}{6}n$$

الحد $\frac{1}{3}n^3$ أكبر بكثير من الحدود الأخرى عندما يكون n كبيراً، لذلك يمكننا القول إن $1^2+2^2+\cdots+n^2$ يساوي تقريباً $\frac{1}{3}n^3$ ، وإذا أردنا أن نكون أكثر دقة، فيمكننا القول إن الفرق بين $1^2+2^2+\cdots+n^2$ و $\frac{1}{3}n^3$ أكبر أو أقل من مضاعفات n^2.

الصيغ التقريبية من هذا النوع تظهر كثيراً في نظرية الأعداد، كغيرها من فروع الرياضيات الأخرى وعلم الحاسب. وهي تأخذ الشكل التالي :

دالة صعبة بدلالة n = دالة سهلة بدلالة n + حد لمقدار الخطأ بدلالة n

على سبيل المثال،

$$\underbrace{1^2+2^2+\cdots+(n-1)^2+n^2}_{\text{دالة صعبة بدلالة } n} = \underbrace{\tfrac{1}{3}n^3}_{\substack{\text{دالة سهلة}\\ \text{بدلالة } n}} + \left(\text{خطأ ليس أكبر بكثير من } n^2\right)$$

الطريقة الرياضية لكتابة هذه الصيغة التقريبية تكون باستخدام الرمز " O كبيرة". باستخدام O كبيرة نكتب الصيغة السابقة كما يلي :

$$1^2+2^2+\cdots+(n-1)^2+n^2=\frac{1}{3}n^3+\mathrm{O}\left(n^2\right)$$

بشكل عام ، هذا يعني أن الفرق بين $1^2+2^2+\cdots+n^2$ و $\frac{1}{3}n^3$ أقل من عدد ثابت مضروب في n^2.

التعريف الرسمي لرمز O كبيرة هو تجريدي نوعاً ما، وقد يسبب بعض الارتباك في البداية. لكن إذا حفظت في عقلك مثال $1^2+2^2+\cdots+n^2$ ، فستجد أن رمز O كبيرة ليس على تلك الدرجة من التعقيد، وبقليل من التمرين سيصبح استخدامه طبيعيًّا جدًّا.

تعريف. لتكن $h(n), g(n), f(n)$ دوال. الصيغة

$$f(n)=g(n)+\mathrm{O}\big(h(n)\big)$$

تعني أن هناك عدداً ثابتاً C وقيمة مبدئية n_0 بحيث

$$|f(n)-g(n)|\le C|h(n)| \quad \forall\ n\ge n_0$$

وبالكلمات ، الفرق بين $f(n)$, $g(n)$ ليس أكبر من عدد ثابت مضروب في $h(n)$. عند قراءة الصيغة $f(n)=g(n)+\mathrm{O}\big(h(n)\big)$ بصوت عالٍ، نقول:

" $f(n)$ تساوي $g(n)$ زائد O كبيرة لـ $h(n)$ "

أحياناً تختفي الدالة $g(n)$ ، وهذا نفس قولنا إن $g(n)=0$ لكل n ؛ لذلك فالصيغة $f(n)=\mathrm{O}\big(h(n)\big)$

تعني أن هناك عدداً ثابتاً C وقيمة مبدئية n_0 بحيث:

$|f(n)|\le C|h(n)| \quad \forall\ n\ge n_0$ ، على سبيل المثال: $n^3=\mathrm{O}(2^n)$ ، لأن[1]:

$$n^3\le 2^n \quad \forall\ n\ge 10$$

---

(١) لاحظ أن هناك العديد من الخيارات الممكنة للعددين C, n_0. على سبيل المثال، يمكننا أن نقول إن $n^3=\mathrm{O}(2^n)$ لأن $n^3\le 10\cdot 2^n\ \forall n\ge 1$. لكن لا نستطيع أن نقول إن $n^3=\mathrm{O}\big(n^2\big)$ لأنه لا يوجد عدد ثابت C يجعل n^3 أقل من Cn^2 عندما يكون n كبيراً.

من الشائع كذلك وجود صيغ يكون فيها $h(n)$ يساوي الدالة الثابتة $h(n)=1$. الخطأ الشائع هو أن تعتقد أن الصيغة

$$f(n)=\mathrm{O}(1)$$

تعني أن $f(n)$ نفسها ثابت. إن الصيغة $f(n)=\mathrm{O}(1)$ تعني أن $|f(n)|$ أقل من ثابت C. على سبيل المثال، الدالة $f(n)=\dfrac{2n+1}{n+2}$ ليست ثابتة، ولكن من الصحيح أن:

$$\frac{2n+1}{n+2}=\mathrm{O}(1)$$

لأن:

$$\left|\frac{2n+3}{n+2}\right|\le 2 \quad \forall \quad n\ge 1$$

متتالية فيبوناتشي:

$$1,1,2,3,5,8,13,\ldots$$

تزودنا بفرصة أخرى لاستخدام رمز O كبيرة. لقد أثبتنا في الفصل السابع والثلاثون صيغة Binet الجميلة لعدد فيبوناتشي n^{th} :

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

المقداران الظاهران في هذه الصيغة لهما القيمتان:

$$\frac{1+\sqrt{5}}{2}=1.618039\cdots \quad , \quad \frac{1-\sqrt{5}}{2}=-0.618039\cdots$$

عندما نأخذ المقدار $\frac{1-\sqrt{5}}{2}$ ونرفعه لقوة كبيرة، سنحصل على مقدار صغير جدًّا، بينما إذا رفعنا المقدار $\frac{1-\sqrt{5}}{2}$ لقوة كبيرة فسنحصل على مقدار كبير جدًّا. لذلك فالصيغة التقريبية- لكنها تبقى مفيدة- لصيغة Binet تقول إن:

$$F_n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n + \mathrm{O}\left(0.61304^n\right)$$

في هذه الحالة، حد الخطأ $\mathrm{O}\left(0.613046^n\right)$ يقترب من الصفر بسرعة كبيرة كلما أصبحت قيمة n أكبر وأكبر. إن هذا مغاير لصيغة O كبيرة للمجموع $1^2+2^2+\cdots+n^2$، حيث إن الخطأ $\mathrm{O}\left(n^2\right)$ يصبح أكبر كلما زادت قيمة n، حتى وإن يكن بمعدل أبطأ من الحد الرئيسي $\frac{1}{3}n^3$.

إن هناك طرقاً كثيرة لاكتشاف وإثبات صيغ O الكبيرة. أحد أكثر هذه الطرق فاعلية تستخدم الهندسة والقليل من حساب التفاضل والتكامل. سنشرح هذه الطريقة الهندسية بإيجاد صيغة O كبيرة للمجموع:

$$1^k+2^k+\cdots+n^k$$

نعلم مسبقاً أن:

$$1+2+\cdots+n=\frac{1}{2}n^2+\mathrm{O}(n)\ ,\ 1^2+2^2+\cdots+n^2=\frac{1}{3}n^3+\mathrm{O}\left(n^2\right)$$

لذلك سنظن أن:

$$1^k+2^k+\cdots+n^k \stackrel{?}{=} \frac{1}{k+1}n^{k+1}+\mathrm{O}\left(n^k\right)$$

وهذا هو الحاصل في الحقيقة.

نظرية (١ ,٣٨):

لتكن $k \geq 1$ قوة ثابتة ، فإن:

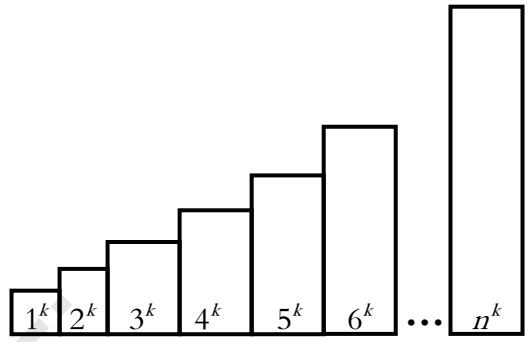
$$1^k + 2^k + 3^k + \cdots + n^k = \frac{1}{k+1} n^{k+1} + \mathrm{O}\left(n^k\right)$$

البرهان:

ليكن $S(n) = 1^k + 2^k + \cdots + n^k$ هو المقدار الذي يعبر عن المجموع الذي نحاول تقدير قيمته. سنرسم حزمة من المستطيلات. المستطيل الأول قاعدته 1 وارتفاعه 1^k والمستطيل الثاني قاعدته 1 وارتفاعه 2^k والمستطيل الثالث قاعدته 1 وارتفاعه 3^k ، وهكذا.

بوضع هذه المستطيلات جنباً إلى جنب (ضلع على ضلع) فسنحصل على الصورة الموضحة في الشكل رقم 38.1. لاحظ أننا إذا جمعنا مساحات هذه المستطيلات فسنحصل على قيمة المقدار $S(n)$.

بدلاً من أن نحسب بالضبط مساحة كل مستطيل من هذه المستطيلات، سوف نقرب المساحة الكلية من خلال حساب مساحة منطقة أبسط. إذا قمنا برسم المنحنى $y = x^k$ ، عندئذ وكما نرى في الشكل 38.2 فإن المستطيلات تقع تحت المنحنى. وبما أن المستطيلات في الشكل رقم (٣٨,٢) تقع تحت المنحنى $y = x^k$ ، فإن مساحة المستطيلات أقل من المساحة تحت المنحنى.



الشكل رقم (٣٨,١). المستطيلات التي مساحتها الكلية تساوي $1^k + 2^k + \cdots + n^k$.

بعبارة أخرى، من الشكل نستنتج أن:

$$1^k + 2^k + \cdots + n^k = \left(\begin{matrix}\text{مساحة}\\ \text{المستطيلات}\end{matrix}\right) < \left(\begin{matrix}\text{المساحة تحت المنحنى } y = x^k\\ \text{حيث } 1 \le x \le n+1\end{matrix}\right)$$

يمكننا استخدام مبادئ التفاضل والتكامل لحساب المساحة تحت المنحنى.

$$\left(\begin{matrix}\text{المساحة تحت المنحنى } y = x^k\\ \text{حيث } 1 \le x \le n+1\end{matrix}\right) = \int_1^{n+1} x^k\,dx$$

$$= \frac{x^{k+1}}{k+1}\Bigg|_1^{n+1}$$

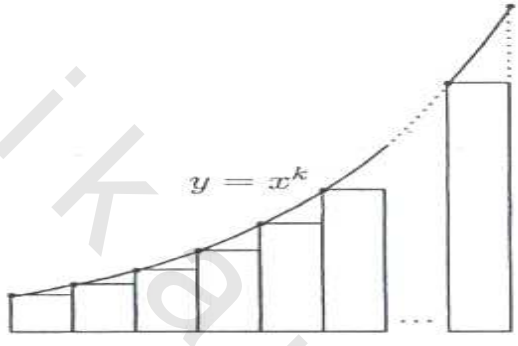
$$= \frac{1}{k+1}\left((n+1)^{k+1} - 1\right)$$

وهذا يعطي الحد الأعلى:

$$1^k + 2^k + \cdots + n^k < \frac{1}{k+1}(n+1)^{k+1}$$

(حذفنا -1 من الطرف الأيمن لنجعل التقدير أفضل).

بنفس الأسلوب، إذا قمنا بتحريك المستطيلات وحدة واحدة لليسار فإن، كما يتضح تماماً من الشكل 38.3، المستطيلات تغطي تماماً المساحة تحت المنحنى $y = x^k$ بين $0 \le x \le n$. هذا يعني أن مساحة المستطيلات أكبر من المساحة تحت المنحنى في هذا الجزء منه؛ لذلك نحصل على الحد الأدنى المناظر:



الشكل رقم (٣٨,٢). المساحة تحت المنحنى أكبر من مساحة المستطيلات.

$$\begin{aligned} 1^k + 2^k + \cdots + n^k &= \left(\begin{matrix}\text{مساحة}\\ \text{المستطيلات}\end{matrix}\right) \\ &> \left(\begin{matrix}\text{المساحة تحت المنحنى } y = x^k \\ \text{حيث } 0 \le x \le n\end{matrix}\right) \\ &= \int_0^n x^k \, dx \\ &= \frac{x^{k+1}}{k+1}\Big|_0^n \\ &= \frac{1}{k+1} \cdot n^{k+1} \end{aligned}$$

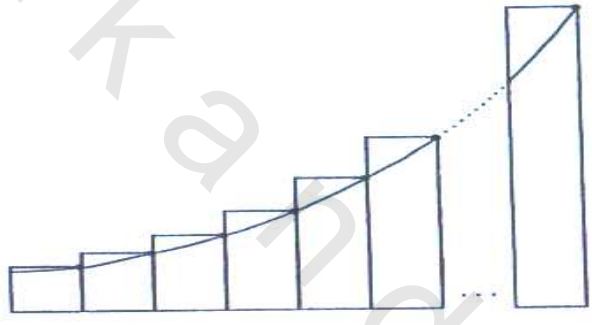
بوضع الحد الأعلى والحد الأدنى مع بعضهما، فإننا نثبت أن:

$$\frac{1}{k+1}\cdot n^{k+1} < 1^k + 2^k + \cdots + n^k < \frac{1}{k+1}(n+1)^{k+1}$$

وبطرح $\frac{1}{k+1}n^{k+1}$ نحصل على:

$$0 < \left(1^k + 2^k + \cdots + n^k\right) - \frac{1}{k+1}\cdot n^{k+1} < \frac{1}{k+1}\left((n+1)^{k+1} - n^{k+1}\right) \ldots (*)$$

الآن نحتاج أن نبين أن الحد الأعلى ليس كبيراً جدًّا. لعمل هذا، نستخدم مفكوك ذو الحدين (الفصل 36) للمقدار $(n+1)^{k+1}$ ،



الشكل رقم (٣٨,٣). المساحة تحت المنحنى أصغر من مساحة المستطيلات .

$$(n+1)^{k+1} = n^{k+1} + \binom{k+1}{1}n^k + \binom{k+1}{2}n^{k-1} + \cdots + \binom{k+1}{k}n + \binom{k+1}{k+1}$$

الجزء الحاسم يكون في أكبر حد وهو n^{k+1} وكل الحدود الأخرى تضم قوى أصغر للعدد n. لذلك:

$$\begin{aligned}(n+1)^{k+1} - n^{k+1} &= \binom{k+1}{1}n^k + \binom{k+1}{2}n^{k-1} + \cdots + \binom{k+1}{k}n + \binom{k+1}{k+1} \\ &\le \binom{k+1}{1}n^k + \binom{k+1}{2}n^k + \cdots + \binom{k+1}{k}n^k + \binom{k+1}{k+1}n^k \\ &= (\text{k حد غامض بدلالة}) \cdot n^k .\end{aligned}$$

بتوليف هذا التقدير مع المتباينة السابقة (*) ينتج :

$$0 < \left(1^k + 2^k + \cdots + n^k\right) - \frac{1}{k+1} \cdot n^{k+1} < n^k \cdot \left(k \quad \textbf{حد غامض بدلالة}\right)$$

طبعاً، "الحد الغامض الأخير" يساوي $\frac{1}{k+1}$ مضروباً في " الحد الغامض السابق"، لكن في جميع الأحوال، هو يضم فقط k ولا يعتمد على n. هذا يثبت أن

$$\left(1^k + 2^k + \cdots + n^k\right) = \frac{1}{k+1} \cdot n^{k+1} + \mathrm{O}\left(n^k\right)$$

وفي الحقيقة فهذا يثبت شيئاً أقوى، حيث إن هذه المعادلة تثبت أن مجموع القوى $1^k + 2^k + \cdots + n^k$ هي دائماً أكبر من $\frac{1}{k+1} n^{k+1}$. هذا يكمل برهان النظرية.

وصف للمدة التي تستغرقها الحسابات

إن رمز O كبيرة يستخدم باستمرار لوصف المدة التي تستغرقها بعض الحسابات باستخدام طريقة معينة. على سبيل المثال، افرض أن لدينا عدداً ثابتاً a ومقياساً m وأردنا حساب قيمة $a^n \pmod m$ لقيمة كبيرة للأس n. كم المدة التي يستغرقها حساب هذه القيمة؟

إحدى هذه الطرق لعمل هذه الحسبة هي حساب،

$a_1 \equiv a \pmod m$ ، ومن ثم

$a_2 \equiv a \cdot a_1 \pmod m$ ، ومن ثم

$a_3 \equiv a \cdot a_2 \pmod m$ ، ومن ثم....

في النهاية نحصل على a_n ، والذي يساوي $a^n \pmod m$.

لقد احتجنا n من الخطوات للوصول إلى هذه النتيجة، وفي كل خطوة ضاعفنا مرة واختزلنا مرة قياس m. لنفرض أن كل خطوة استغرقت أقل أو أكثر من زمن

محدد، وبالتالي سيكون الزمن الكلي من مضاعفات n. الزمن المستغرق في هذه الطريقة هو $O(n)$.

طبعاً، لا يوجد أحد من الذين فرأوا الفصل السادس عشر سيستخدم هذه الطريقة السخيفة لحساب $a^n \pmod m$. إن طريقة التربيع المتتالي تمكننا من حساب $a^n \pmod m$ بشكل أسرع. كما تعلمنا في الفصل السادس عشر. طريقة التربيعات المتتالية لها ثلاث خطوات:

1. كتابة n كمجموع قوى للعدد 2 (المفكوك الثنائي)

$$n = u_0 + u_1 \cdot 2 + u_2 \cdot 4 + u_3 \cdot 8 + \cdots + u_r \cdot 2^r$$

حيث $u_r = 1$ وكل u_i تساوي 0 أو 1.

2. إنشاء جدول بالقيم:

$$A_0 = a, \quad A_1 = A_0^2 \pmod m, \quad A_2 = A_1^2 \pmod m,$$
$$A_3 = A_2^2 \pmod m, \quad \ldots, \quad A_r = A_{r-1}^2 \pmod m.$$

3. حساب حاصل الضرب:

$$A_0^{u_0}, A_1^{u_1}, A_2^{u_2}, \ldots, A_r^{u_r} \pmod m \qquad \ldots\ldots\ldots\ldots\ldots\ldots (38.1)$$

كل خطوة من هذه الخطوات الثلاث تحتاج تقريباً إلى r من الخطوات، إذن الزمن الكلي من مضاعفات العدد r. هل لاحظت ماهية العلاقة بين r والأس n؟ من المفكوك الثنائي (38.1)، العدد n يساوي على الأقل 2^r؛ لذلك إذا أخذنا اللوغاريتم سنرى أن[1].

---

(١) الدالة $\log_2(x)$ هي لوغاريتم أساسه 2. من تعريف اللوغاريتم، قيمة $\log_2(x)$ هي العدد y الذي يحقق المعادلة $2^y = x$.

$$r \le \log_2(n)$$

لذلك ؛ طريقة التربيعات المتتالية تمكننا من حساب $a^n \pmod m$ في زمن يساوي $O(\log_2(n))$. وهذا أسرع بكثير جدًّا من الزمن $O(n)$ ؛ لأن $\log_2(n)$ أصغر بكثير من n عندما يكون n كبيراً.

العدد $\log_2(n)$ هو عدد الخانات الثنائية في n ، أي هو عدد الخانات عندما نكتب n باستخدام رمز ثنائي. بالمثل، $\log_{10}(n)$ هو عدد الخانات العشرية في n. لذلك، بكلام تقريبي، $\log(n)$ تخبرنا المعلومات التالية :

- كم يُستغرق من الوقت لمعرفة العدد n.
- كم نحتاج من الوقت لوصف العدد n لشخص آخر.
- كم نحتاج من الوقت لإدخال العدد n إلى الكمبيوتر أو لظهور العدد n كمخرج من الكمبيوتر.

يمكننا تلخيص ذلك بقولنا إن الوقت المُستغرق هو $O(\log(n))$ لوصف العدد n.

من المثير للاهتمام والاستغراب أن الوقت المُستغرق في حساب المقدار $a^n \pmod m$ هو من مضاعفات $O(\log(n))$ ؛ لأننا رأينا أنها تستغرق $O(\log(n))$ من الوقت بمجرد إدخال العدد n. نقول إن طريقة التربيع المتتالي تستغرق "زمناً خطيًّا" لأن عدد القيم المضروبة يكون على الأكثر ثابتاً مضروباً في الوقت الذي نحتاجه لإدخال المعلومة المبدأية.

(طبعاً، إذا كان m , n من نفس الحجم، عندئذ كل عملية ضرب تحتاج على الأقل $(O(\log(n)))$ خطوة ؛ لذلك فإن الزمن الكلي يساوي على الأقل $O(\log(n)^2)$).

دعنا نلقي نظرة على مسألة أخرى، وهي مسألة ضرب كثيري حدود من الدرجة d ،

$$F(X)=a_0+a_1X+\cdots+a_dX^d \ , \ G(X)=b_0+b_1X+\cdots+b_dX^d$$

إن هناك $2d+2$ معامل؛ لذلك الوقت المستغرق هو $\mathrm{O}(d)$ لوصف كثيرات الحدود[1].

ضرب الدالة $F(X)$ و $G(X)$ يُعطى بالصيغة:

$$H(X)=F(X)G(X)=c_0+c_1X+c_2X^2+\cdots+c_{2d}X^{2d}$$

حيث:

$$c_j=\begin{cases} a_0b_j+a_1b_{j-1}+\cdots+a_jb_0 & , \quad if \quad 0\le j\le d \\ a_{j-d}b_d+a_{j-d+1}b_{d-1}+\cdots+a_db_{j-d} & , \quad if \quad d<j\le 2d \end{cases}$$

إذا كان $0\le j\le d$ ، فإن حساب c_j يتطلب j عملية جمع و $j+1$ عملية ضرب؛ لذلك فهذه الحسبة تستغرق $\mathrm{O}(j)$ من الوقت. وإذا كان $d<j\le 2d$ فإن هذه الحسبة تستغرق $\mathrm{O}(2d-j+1)$ من الوقت؛ لذلك الزمن الكلي لحساب الضرب $H(X)$ يساوي:

---

(١) لاحظ أن الدور الذي تلعبه درجة كثير الحدود هو نفس الدور الذي يلعبه لوغاريتم عدد. خاصية أخرى تشترك فيها الدرجة واللوغاريتم توضح من خلال المعادلات:

$$\deg\big(F(X)G(X)\big)=\deg\big(F(X)\big)+\deg\big(G(X)\big)$$
$$\log(MN)=\log(M)+\log(N)$$

أي أن الدرجة واللوغاريتم كليهما يحول الضرب إلى جمع.

$$\sum_{j=0}^{d} \mathrm{O}(j) + \sum_{j=d+1}^{2d} \mathrm{O}(2d-j) = \mathrm{O}\left(\sum_{j=0}^{d} (j) + \sum_{j=d+1}^{2d} (2d-j+1) \right)$$

$$= \mathrm{O}\left(\frac{d^2+d}{2} + \frac{d^2+d}{2} \right) = \mathrm{O}\left(d^2\right)$$

إذن الوقت الذي نحتاجه لحساب $F(X)G(X)$ يساوي O كبيرة لمربع مقدار الوقت اللازم لإدخال البيانات المبدأية ؛ لذلك نقول إنها تستغرق مربع الوقت لحساب ضرب كثيري حدود[١] .

تمارين

(٣٨,١) (a) افرض أن :

$$f_1(n) = g_1(n) + \mathrm{O}\left(h(n)\right) \quad , \quad f_2(n) = g_2(n) + \mathrm{O}\left(h(n)\right)$$

برهن أن

$$f_1(n) + f_2(n) = g_1(n) + g_2(n) + \mathrm{O}\left(h(n)\right)$$

(b) بشكل عام، إذا كان a , b أي عددين ثابتين، برهن أن

$$a f_1(n) + b f_2(n) = a g_1(n) + b g_2(n) + \mathrm{O}\left(h(n)\right)$$

(لاحظ أن الثابت C الذي يظهر في تعريف رمز O الكبيرة مسموح أن يعتمد على الثابتين a,b . المطلوب فقط هو أن تكون هناك قيمة ثابتة واحدة لـ C

---

(١) بالطبع نحن نعني أن هذه الطريقة لحساب $F(X)G(X)$ تستغرق مربع الوقت. هناك طرق أخرى، مثل مضاعف Karatsuba وتحويلات Fourier السريعة، والتي تعتبر أسرع. هذه الطرق قادرة على ضرب كثيري حدود في وقت $\mathrm{O}(d \log d)$ ؛ لذلك هي أبطأ بقليل من الزمن الخطي. ما يميز الزمن الخطي عن الزمن التربيعي ليس على درجة كبيرة من الأهمية إذا كان d صغيراً، مثل $d = 10$ أو $d = 15$ ، لكن إذا كان $d = 10000$ ، فهناك ميزة معتبرة.

تصلح لأي قيمه كبيرة بما فيه الكفاية لـ n).

(c) الصيغة التي قمت بإثباتها في الفقرة (b) تبين أن صيغ O الكبيرة (التي لها نفس h) يمكن جمعها، طرحها، وضربها بعدد ثابت. هل يمكن ضرب هذه الصيغ بمقادير ليست ثابتة؟ بمعنى آخر، إذا كان $f(n)=g(n)+\mathrm{O}\big(h\big)(n)$ وإذا كان $k(n)$ دالة أخرى في n ، هل صحيح أن:

$$k(n)f(n)=k(n)g(n)+\mathrm{O}\big(h(n)\big)\ ?$$

إذا لا، فهل صحيح أن:

$$k(n)f(n)=k(n)g(n)+\mathrm{O}\big(k(n)h(n)\big)\ ?$$

(٣٨,٢) افرض أن:

$$f_1(n)=g_1(n)+\mathrm{O}\big(h_1(n)\big)\quad,\quad f_2(n)=g_2(n)+\mathrm{O}\big(h_2(n)\big)$$

برهن أن

$$f_1(n)+f_2(n)=g_1(n)+g_2(n)+\mathrm{O}\big(\max\{h_1(n),h_2(n)\}\big)$$

(٣٨,٣) أي من هذه الدوال هو $\mathrm{O}(1)$ ؟ لماذا؟

$$(a)\ f(n)=\frac{3n+17}{2n-1}\qquad (b)\ f(n)=\frac{3n^2+17}{2n-1}$$

$$(c)\ f(n)=\frac{3n+17}{2n^2-1}\qquad (d)\ f(n)=\cos(n)$$

$$(e)\ f(n)=\frac{1}{\sin\left(1/n\right)}\qquad (f)\ f(n)=\frac{1}{n\cdot\sin\left(1/n\right)}$$

(٣٨,٤) أوجد تقدير O كبيرة لمجموع جذور تربيعية، بمعنى آخر، املأ الفراغ في الصيغة التالية $\sqrt{1}+\sqrt{2}+\sqrt{3}+\cdots+\sqrt{n}=\square\, n^{\square}+\mathrm{O}\big(n^{\square}\big)$

(٣٨,٥) (a) برهن تقدير O كبيرة التالي لمجموع مقلوبات الأعداد الصحيحة :

$$1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\frac{1}{5}+\cdots+\frac{1}{n}=\ln(x)+\mathrm{O}(1)$$

[$\ln(x)$ هنا هو اللوغاريتم الطبيعي في x].

(b) برهن العبارة الأقوى وهي أنه يوجد ثابت γ بحيث :

$$1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\frac{1}{5}+\cdots+\frac{1}{n}=\ln(x)+\gamma+\mathrm{O}\left(\frac{1}{n}\right)$$

العدد γ والذي يساوي $0.577215664\cdots$ يسمى ثابت أويلر. قليلون هم الذين يعرفون عن ثابت أويلر. فمثلاً ، ليس معروفاً فيما إذا كان γ عدداً نسبيًّا أم لا.

(٣٨,٦) يلعب "بوب" و "ألس" لعبة التخمين التالية. تقوم ألس باختيار عدد في سرها (أي بدون أن يعلم به بوب) بين 1 و n. يبدأ بوب بتخمين ذلك العدد، وتقوم ألس بعد كل تخمين بإخبار بوب فيما إذا كان تخمينه صحيحاً أم لا. ليكن $G(n)$ هو الحد الأقصى لعدد تخمينات بوب الذي يحتاجه لتخمين عدد ألس، وافرض أن بوباً استخدم أفضل إستراتيجية تخمين ممكنة.

(a) برهن أن $G(n)=\mathrm{O}(n)$.

(b) برهن أن $G(n)$ لا يساوي $\mathrm{O}(\sqrt{n})$.

(c) بشكل عام، إذا كان $G(n)=\mathrm{O}(h(n))$ ، ماذا تستطيع أن تقول عن الدالة $h(n)$ ؟

(d) افرض أننا غيرنا قوانين اللعبة بحيث أصبحت ألس تخبر بوباً بعد كل تخمين فيما إذا كان تخمينه أكبر، أصغر، أو هو بالضبط. صِفْ إستراتيجية لبوب بحيث يكون عدد تخميناته قبل أن يفوز يحقق المعادلة $G(n)=\mathrm{O}(\log_2(n))$.

(مساعدة: إلغِ نصف الأعداد المتبقية في كل تخمين).

(٣٨,٧) يعلم بوب أن العدد n غير أولي ويريد إيجاد أحد عوامله غير 1 أو n. اتبع بوب الإستراتيجية التالية: يفحص فيما إذا كان 2 يقسم n، ثم يفحص فيما إذا كان 3 يقسم n، ثم يفحص فيما إذا كان 4 يقسم n، وهكذا. ليكن $F(n)$ عدد الخطوات التي استغرقها بوب حتى وجد عاملاً من عوامل n.

(a) برهن أن $F(n)=\mathrm{O}(\sqrt{n})$.

(b) افرض أن، بدلاً من فحص كل رقم $2,3,4,5,6,\ldots$، اختير بوب فقط فيما إذا كان n يقبل القسمة على الأعداد الأولية $2,3,5,7,11,\ldots$. اشرح لماذا تظل هذه الإستراتيجية صحيحة، وبيّن أن عدد الخطوات $F(n)$ في هذه الإستراتيجية يحقق $F(n)=\mathrm{O}\left(\frac{\sqrt{n}}{\ln(n)}\right)$. (مساعدة: ستحتاج لاستخدام نظرية العدد الأولي 13.1). هل تعتقد أن هذه الإستراتيجية الجديدة فعّالة؟

(c) من الطرق الأسرع والمعروفة لحل هذه المسألة طريقة "المنخل التربيعي" وطريقة "المنحنى الناقصي". عدد الخطوات $L(n)$ التي تتطلبها هذه الطرق يحقق:

$$L(n)=\mathrm{O}\left(e^{c\sqrt{\ln(n)\cdot\ln\ln(n)}}\right)$$

حيث c عدد ثابت صغير. برهن أن هذه الطرق أسرع من الطريقة الواردة في فقرة (a) من خلال برهان أن:

$$\lim_{n\longrightarrow\infty}\frac{e^{c\sqrt{\ln(n)\cdot\ln\ln(n)}}}{\sqrt{n}}=0$$

بشكل أعم، بيّن أن النهاية تساوي صفراً حتى لو استبدلنا $\sqrt{n}$ الموجود في المقام بالمقدار n^{ϵ} حيث $\epsilon>0$.

(d) أسرع طريقة معروفة لحل هذه المسألة لأعداد كبيرة n تسمى "منخل حقل العدد" (NFS). عدد الخطوات $M(n)$ اللازمة في طريقة NFS تساوي:

$$M(n)=\mathrm{O}\left(e^{c'\sqrt[3]{(\ln n)(\ln\ln n)^2}}\right)$$

حيث مرة أخرى c' عدد ثابت صغير. برهن أنه لقيم كبيرة لـ n فإن الدالة $M(n)$ أصغر بكثير من تقدير O الكبيرة لـ $L(n)$ في الفقرة (c).

رمز O الكبيرة قدم فائدة للرياضيين وعلماء الكمبيوتر بابتكار رمز مشابه لوصف بعض الحالات المشابهة الأخرى. في التمارين القليلة القادمة، سوف نقدم لبعض هذه المفاهيم ونطلب منك تقديم بعض الأمثلة.

(٣٨,٨) **رمز o صغيرة.** من البديهي أن الرمز $o(h(n))$ يشير إلى كمية أصغر بكثير من $h(n)$. **التعريف الدقيق هو:**

$$f(n)=g(n)+o(h(n)) \quad \textbf{أي} \quad \lim_{n\longrightarrow\infty}\frac{f(n)-g(n)}{h(n)}=0$$

(a) برهن أن $n^{10}=o(2^n)$.

(b) برهن أن $2^n=o(n!)$.

(c) برهن أن $n!=o\left(2^{n^2}\right)$.

(d) ماذا تعني الصيغة $f(n)=o(1)$؟ أي من الدوال التالية تكون $o(1)$؟

$$(i)\ f(n)=\frac{1}{\sqrt{n}} \qquad (ii)\ f(n)=\frac{1}{\sin(n)} \qquad (iii)\ f(n)=2^{n-n^2}$$

(٣٨,٩) **رمز أوميجا الكبيرة.** رمز أوميجا الكبير يشبه كثيراً رمز O كبيرة، إلا أن المتباينة هنا تكون مقلوبة[1]، بمعنى:

---

(١) تحذير: التمرين 38.9 يصف ماذا يعني الرمز Ω لعلماء الكمبيوتر. المعنى مختلف بالنسبة للرياضيين. يعني للرياضيين وجود عدد ثابت موجب C وعدد لا نهائي من القيم لـ n بحيث $|f(n)-g(n)|\geq C|h(n)|$. لاحظ الفرق المهم بين جملة تكون صحيحة لكل القيم (الكبيرة) لـ n وأنها تكون صحيحة لعدد لا نهائي من القيم لـ n.

$$f(n) = g(n) + \Omega\big(h(n)\big)$$

هذا يعني أنه يوجد عدد ثابت C وقيمة ابتدائية n_0 بحيث:

$$\big|f(n) - g(n)\big| \geq C\big|h(n)\big| \quad \text{لكل } n \geq n_0$$

إذا كان g يساوي صفراً، ففي هذه الحالة يكون $f(n) = \Omega\big(h(n)\big)$ أي $\big|f(n)\big| \geq C\big|\big(h(n)\big)\big|$ لكل قيمة كبيرة بما يكفي لـ n.

(a) برهن أن كل صيغة من الصيغ التالية هي صيغة صحيحة:

$$(i)\ n^2 - n = \Omega(n) \qquad (ii)\ n! = \Omega\big(2^n\big) \qquad (iii)\ \frac{5^n - 3^n}{2^n} = \Omega\big(2^n\big)$$

(b) إذا كان $f(n) = \Omega\big(h(n)\big)$ و $h(n) = \Omega\big(k(n)\big)$.

برهن أن $f(n) = \Omega\big(k(n)\big)$

(c) إذا كان $f(n) = \Omega\big(h(n)\big)$، فهل من الصحيح دائماً أن $h(n) = \mathrm{O}\big(f(n)\big)$؟

(d) ليكن $f(n) = n^3 - 3n^2 + 7$. ما هي قيم d التي تجعل الصيغة $f(n) = \Omega\big(n^d\big)$ صحيحة؟

(e) ما هي قيم d التي تجعل الصيغة $\sqrt{n} = \Omega\Big(\big(\log_2 n\big)^d\Big)$ صحيحة؟

(f) برهن أن الدالة $f(n) = n \cdot \sin(n)$ لا تحقق $f(n) = \Omega\big(\sqrt{n}\big)$.

[مساعدة: استخدم نظرية ديرتشلت للتقريب الديوفانتيني 31.2 لإيجاد كسور p/q تحقق $|p - 2\pi q| < 1/q$، افرض $n = p$، واستخدم حقيقة أن $\sin(x) \approx x$ عندما تكون x صغيرة].

(٣٨.١٠) **رمز ثيتا الكبيرة**. رمز ثيتا الكبيرة يجمع بين O الكبيرة و Ω الكبيرة. إحدى طرق تعريف ثيتا الكبيرة هو استخدام التعريفات السابقة ونقول إن:

$$f(n) = g(n) + \Theta\big(h(n)\big)$$

إذا كان :

$$f(n)=g(n)+\Omega\big(h(n)\big) \quad , \quad f(n)=g(n)+\mathrm{O}\big(h(n)\big)$$

أو يمكننا كتابة كل شيء بشكل صريح ونعرف

$$f(n)=g(n)+\Theta\big(h(n)\big)$$

لتعني أنه يوجد عددان ثابتان موجبان C_1, C_2 وقيمة ابتدائية n_0 بحيث

$C_1|h(n)|\le|f(n)-g(n)|\le C_2|h(n)|$ لكل $n\ge n_0$

(a) برهن أن :

$$\ln\left(1+\frac{1}{n}\right)=\Theta\left(\frac{1}{n}\right)$$

[مساعدة : استخدم مفكوك متسلسلة تايلور لـ $\ln(1+t)$ لتقدير قيمته عندما يكون t صغيراً].

(b) استخدم فرع (a) لإثبات أن :

$$\ln|n^3-n^2+3|=3\ln(n)+\Theta\left(\frac{1}{n}\right)$$

(c) عمم (b) وأثبت أنه إذا كان $f(x)$ كثير حدود من الدرجة d فإن

$$\log|f(n)|=d\ln(n)+\Theta\left(\frac{1}{n}\right)$$

(d) إذا كان $f_1(n)=g_1(n)+\Theta\big(h(n)\big)$, $f_2(n)=g_2(n)+\Theta\big(h(n)\big)$ ، برهن أن $f_1(n)=f_2(n)=g_1(n)+g_2(n)+\Theta\big(h(n)\big)$.

(c) إذا كان $f(n)=\Theta\big(h(n)\big)$ ، فهل من الضروري أن تكون المعادلة $h(n)=\Theta\big(f(n)\big)$ صحيحة؟