ДОМАШНО № 1 ПО ДИЗАЙН И АНАЛИЗ НА АЛГОРИТМИ

С О Р Т И Р А Н Е И Т Ъ Р С Е Н Е

Указания:
1) Всички решения да бъдат обосновани подробно.
2) Алгоритмите, изучени на лекции, могат да се използват наготово.
3) При всяко сортиране и търсене да се уточнява използваният алгоритъм.
4) Названията на всички алгоритми и структури от данни трябва да бъдат на български език.

Задача 1. Двама души A и B играят един срещу друг по следните правила. Дадено е цяло число $N > 1$. Играчът A избира цяло число X от 1 до N вкл. Играчът B трябва да познае числото X с помощта на въпроси от следния вид: "Вярно ли е, че X е по-голямо или равно на K ?", където K е естествено число, избирано от играча B. На всеки въпрос играчът A отговаря правилно (не лъже) и не променя числото X по време на играта. Играчът B плаща на играча A за всеки отговор: по два лева за отговор "да" и по един лев за отговор "не".

За дадено число N да се определи най-малко колко лева са достатъчни, за да се гарантира отгатването на което и да е цяло число X от 1 до N вкл. Да се състави оптимална стратегия за играча B и да се демонстрира подробно при $N = 10$ и $X = 7$.

Упътване: Трудно е да се намери направо търсената функция на N. По-лесно е да се реши отначало обратната задача: ако играчът B има L лева, да се намери най-голямото N, такова че тези L лева гарантират на B отгатването на всяко цяло число X от 1 до N включително.

Задача 2. Дадени са n предмета и два масива — $W[1 \dots n]$ и $S[1 \dots n]$. Числото $W[k]$ е теглото на k-тия предмет, а $S[k]$ — неговата издръжливост. Издръжливост на един предмет наричаме най-голямото тегло, което може да се сложи върху предмета, без той да се повреди (да се смачка или счупи). Всички числа са положителни (цели или дробни). Можем да си мислим, че предметите са някакви покупки, които трябва да се наредят една върху друга (например в торба за пазаруване) така, че да не се повредят.

Съставете алгоритъм, който подрежда предметите за време $O(n \log n)$ при всякакви входни данни (или съобщава, че те не могат да се подредят).

ТОЧКУВАНЕ

Представените решения на задачите от домашното се оценяват в проценти. Пълни решения на двете задачи носят общо 100 % — по 50 % за всяка задача.

РЕШЕНИЯ

Задача 1 е дадена на Първата международна олимпиада по програмиране, проведена в Чехословакия през август 1987 г. Играта, описана в условието, е разновидност на двоичното търсене, но нейният анализ е доста по-сложен.

Нека T(N) е търсената минимална парична сума, гарантираща отгатването на всяко цяло число от 1 до N включително. Ясно е, че играчът B няма полза от въпроси с предварително известен отговор (това е чиста загуба на пари). Затова K е естествено число от 2 до N вкл. Правилото за плащането означава, че функцията T(N) удовлетворява следното рекурентно уравнение:

$$\mathrm{T}(N) = \min_{K} \max \{\, 2 + \mathrm{T}(N - K + 1)\, ;\ 1 + \mathrm{T}(K - 1) \,\} \quad \text{за всяко } N > 2,$$

където елементите на множеството съответстват на двата възможни отговора на първия въпрос, зададен от играча B; максимумът от двете възможности се взима заради най-лошия случай за X (тъй като се иска гарантирано отгатване на всяко допустимо число X); по допустимите стойности на K е взет минимум, тъй като играчът B сам избира числото K (а числото X се избира от играча A). Първият член на множеството се основава на факта, че паричната сума T(N) е една и съща, независимо за кои N числа става дума (може да не са от 1 до N), стига да са последователни и интервалът да е известен на играча B.

Тези разсъждения са естествени: все едно анализираме времевата сложност на алгоритъм (играта на въпроси и отговори), като всяка операция (въпрос) отнема една или две единици време в зависимост от резултата (отговора) — съответно “не” или “да”. От друга страна, този подход, макар и естествен, ни доведе до трудно уравнение. Затова обръщаме задачата: вместо по дадено N да търсим най-малката парична сума L = T(N), която гарантира отгатването на всяко цяло число X от 1 до N включително, ще се опитаме да отговорим на обратния въпрос: ако разполагаме с L лева, кое е най-голямото цяло число N, за което L лева е най-малката парична сума, гарантираща отгатването на всяко цяло число X от 1 до N включително? Да означим въпросното число N с A(L). Функцията N = A(L) в известен смисъл е обратна на функцията L = T(N). “В известен смисъл”, защото функцията T може да не е биекция, поради което може да няма обратна функция. По-точно, връзката между двете функции е следната: L = T(N) е единственото цяло число, за което $\mathrm{A}(L - 1) < N \leq \mathrm{A}(L)$. Ето защо можем да намерим функцията T(N), ако знаем функцията N = A(L).

Следователно играчът B трябва да избере първото число K така, че:

— ако получи отговор “да” (за който плаща два лева), да може да отгатне всяко X измежду числата K, $K + 1$, $K + 2$, . . . , N (общо $N - K + 1$ на брой), като плати допълнително не повече от $L - 2$ лева;

— ако получи отговор “не” (за който плаща един лев), да може да отгатне всяко X измежду числата 1, 2, 3, . . . , $K - 1$, доплащайки не повече от $L - 1$ лева.

От двете изисквания следва, че $N - K + 1 \leq \mathrm{A}(L - 2)$ и $K - 1 \leq \mathrm{A}(L - 1)$.

Поради изискването за максималност в определението на функцията А(L) е изпълнено равенство и на двете места:

$$N - K + 1 = \mathrm{A}(L - 2) \text{ и } K - 1 = \mathrm{A}(L - 1).$$

След събиране на двете равенства стигаме до извода, че $N = \mathrm{A}(L - 1) + \mathrm{A}(L - 2)$. Понеже N = А(L), то за редицата А(L) се получава рекурентното уравнение

$$\mathrm{A}(L) = \mathrm{A}(L - 1) + \mathrm{A}(L - 2) \text{ за всяко } L \geq 2.$$

За да определим редицата А(L) напълно, ни трябват две начални условия. Очевидно А(0) = 1 и А(1) = 1. Следователно А(L) е редицата на Фибоначи. Както знаем,

$$N = \mathrm{A}(L) = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^{L+1},$$

като закръглянето е по обичайния начин — към най-близкото цяло число (оказва се, че закръглянето е нагоре при четно L и надолу при нечетно L).

Като решим това уравнение относно L, намираме

$$L = \mathrm{T}(N) = \left\lceil \log_{\frac{1+\sqrt{5}}{2}}\left(\sqrt{5}\, N\right) \right\rceil - 1,$$

като трябва да извадим още една единица (тоест да заменим умалителя 1 с 2), ако N е число на Фибоначи с четен индекс, тоест 1, 2, 5, 13, 34 и тъй нататък (първите два числа на Фибоначи са единици и индексирането започва от 0). Полученият израз за Т(N) е равен на най-малката парична сума, достатъчна за отгатването на всяко цяло число от 1 до N включително.

От формулата $K - 1 = \mathrm{A}(L - 1)$ намираме $K = 1 + \mathrm{A}(L - 1)$. Това равенство задава оптималната стратегия: играчът B трябва да закръгли даденото число N нагоре до най-близкото по-голямо или равно на N число на Фибоначи А(L) и да зададе въпрос с $K = 1 + \mathrm{A}(L - 1)$. Тази стратегия гарантира отгатване с минимална цена: най-много L лева.

На практика стратегията препоръчва играчът B винаги да работи с интервал, чиято дължина е число на Фибоначи; да разделя интервала на два подинтервала, чиито дължини са двете предходни числа на Фибоначи; левият подинтервал да бъде по-дълъг от десния; играчът B трябва да зададе въпрос с онова число K, което е левият край на десния подинтервал.

П р и м е р : Нека $N = 10$. Първите няколко числа на Фибоначи са следните: А(0) = 1, А(1) = 1, А(2) = 2, А(3) = 3, А(4) = 5, А(5) = 8, А(6) = 13 и т.н. Най-малкото число на Фибоначи, което е по-голямо или равно на $N = 10$, е числото А(6) = 13. Неговият индекс е 6, което означава, че Т(10) = 6 лева, тоест 6 лева е най-малката парична сума, която гарантира точното отгатване на всяко цяло число от 1 до 10 включително.

На първия си ход играчът B разбива интервала $\{1 ; 2 ; 3 ; \ldots ; 12 ; 13\}$ на два подинтервала с дължини 8 и 5 — предходните две числа на Фибоначи. Левият подинтервал трябва да съдържа осем числа, тоест $\{1 ; 2 ; 3 ; \ldots ; 7 ; 8\}$. Десният подинтервал трябва да съдържа пет числа, тоест $\{9 ; 10 ; 11 ; 12 ; 13\}$. За първия си въпрос играчът B избира левия край на десния подинтервал, тоест числото $K = 9$. С други думи, B задава въпроса: "Вярно ли е, че $X \geq 9$?"

Нека играчът A е избрал например числото $X = 7$. Тогава отговорът е "не" и B плаща един лев на A. След този отговор B знае, че числото X принадлежи на левия подинтервал $\{1 ; 2 ; 3 ; \ldots ; 7 ; 8\}$.

На втория си ход B разбива този интервал на подинтервали с дължини 5 и 3: $\{1 ; 2 ; 3 ; 4 ; 5\}$ и $\{6 ; 7 ; 8\}$. Ето защо $K = 6$ и B пита: "Вярно ли е, че $X \geq 6$?" Отговорът е "да" и B плаща два лева на A. Сега B знае, че $X \in \{6 ; 7 ; 8\}$.

На третия си ход B разбива този интервал на подинтервали с дължини 2 и 1: $\{6 ; 7\}$ и $\{8\}$. Числото $K = 8$, тоест B задава въпроса: "Вярно ли е, че $X \geq 8$?" Отговорът е "не" и B плаща един лев на A. Сега B знае, че $X \in \{6 ; 7\}$.

На четвъртия си ход B разбива този интервал на подинтервали с дължини 1: $\{6\}$ и $\{7\}$. Сега числото $K = 7$, т.е. B задава въпроса: "Вярно ли е, че $X \geq 7$?" Отговорът е "да" и B плаща два лева на A. В този миг B знае, че $X = 7$, тоест числото е отгатнато.

Цялата игра струва $1 + 2 + 1 + 2 = 6$ лева на играча B, както се очакваше въз основа на равенството $\mathrm{T}(10) = 6$.

От свойствата на числата на Фибоначи следва, че отношението, в което играчът B дели даден интервал, съвпада приблизително със златното сечение.

Задача 2 е от изпит в Оксфорд. Решаваме я, като подредим предметите по сбора $W[k] + S[k]$: тези с голям сбор слагаме отдолу — на дъното на чантата. Интуитивното обяснение е, че долните предмети трябва да са по-издръжливи, а горните — по-леки (тоест по-тежките предмети трябва да отидат на дъното). Сборовете $W[k] + S[k]$ пресмятаме за време $\Theta(n)$ с едно обхождане на масивите, а предметите подреждаме за време $\Theta(n \log n)$ чрез някоя бърза сортировка, например с пирамидалното сортиране.

Коректност на алгоритъма: Наричаме допустими наредбите на предметите, при които никой предмет не се поврежда. Може да няма допустими наредби. Но ако има допустими наредби, ще докажем, че измежду тях съществува такава с ненамаляващи сборове $W[k] + S[k]$. Идеята е проста: трябва да докажем, че ако в допустима наредба има два поредни предмета (да кажем, № k и № k–1), за които $W[k-1] + S[k-1] > W[k] + S[k]$, то след разместването им се получава пак допустима наредба. Повтаряйки това действие, рано или късно ще получим наредба с ненамаляващи сборове "тегло + издръжливост". Всъщност прилагаме метода на мехурчето; той е бавен за практически цели, но е много удобен за съставяне на доказателства в теорията.

Доказателство, че след разместване от описания вид се получава отново допустима наредба: Предметите, чиито номера са различни от k и $k–1$, няма да бъдат повредени след разместването, тъй като носят същата тежест (а наредбата преди разместването е допустима според нашето предположение). Предметът с бивш номер k също няма да се повреди: след разместването се намира на място № $k–1$, т.е. носи по-малка тежест. Опасност от повреждане има само за предмета с номер k след разместването (той е имал номер $k–1$ преди разместването): след разместването този предмет носи по-голяма тежест. Ще докажем, че той може да издържи тази тежест, тоест

$$S[k-1] \geq W[1] + W[2] + \ldots + W[k-3] + W[k-2] + W[k]$$

(всички индекси се отнасят за местата на предметите преди разместването).

Действително, щом наредбата преди разместването е допустима, то

$$S[k] \geq W[1] + W[2] + \ldots + W[k-3] + W[k-2] + W[k-1].$$

Следователно

$$S[k] - W[k-1] \geq W[1] + W[2] + \ldots + W[k-3] + W[k-2]. \qquad (1)$$

От друга страна,

$$W[k-1] + S[k-1] > W[k] + S[k],$$

затова

$$S[k-1] - W[k] > S[k] - W[k-1]. \qquad (2)$$

От транзитивността на неравенствата (1) и (2) заключаваме, че

$$S[k-1] - W[k] > W[1] + W[2] + \ldots + W[k-3] + W[k-2],$$

откъдето след прехвърляне на $W[k]$ следва желаното неравенство.

ПОДРОБНА СХЕМА ЗА ТОЧКУВАНЕ

Задача 1 носи общо 50 %, разпределени по следния начин:
— за съставянето на рекурентно уравнение за $A(L)$: 20 %;
— за намирането на две начални условия за $A(L)$: 6 % (по 3 % за всяко от тях);
— за извода, че членовете на редицата $A(L)$ са числата на Фибоначи: 4 %;
— за намирането на $T(N)$: 5 %;
— за описание на оптималната стратегия на играча B: 5 %;
— за подробна демонстрация на стратегията: 10 %.

Задача 2 носи общо 50 %, разпределени по следния начин:
— за избор на ключ ($W[k] + S[k]$), по който да се сортират предметите: 15 %;
— за избор на конкретна сортировка с времева сложност $O(n \log n)$: 5 %;
— за формулиране на обща идея за доказателството за коректност: 15 %;
— за подробно формално доказателство за коректност: 15 %.