

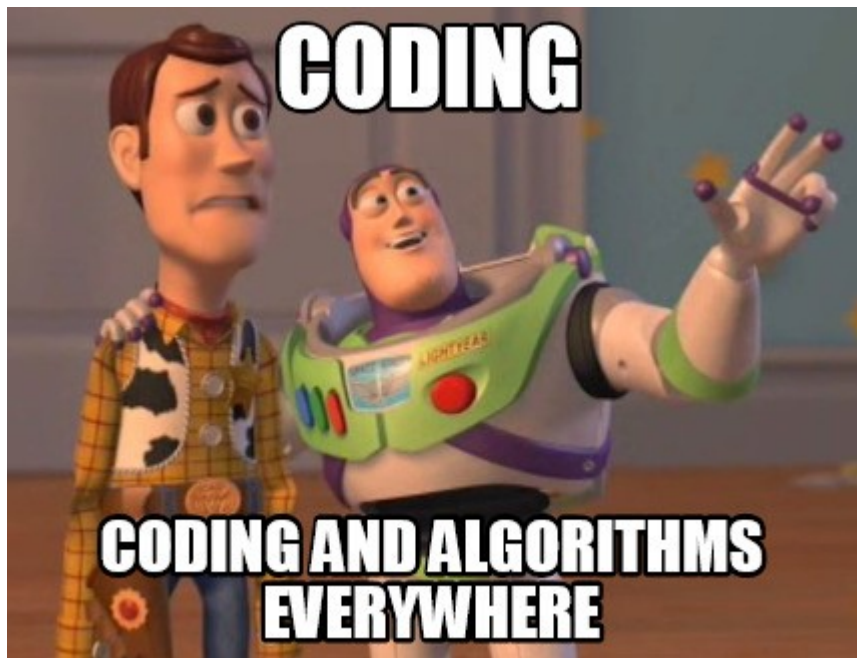
УП Семинар 8

Алгоритми. Сложност на алгоритъм. Двоично търсене. Двоично търсене по отговора.



Проба за дефиниция на понятието алгоритъм.

- Последователност от елементарни стъпки за решаване на по-сложна задача

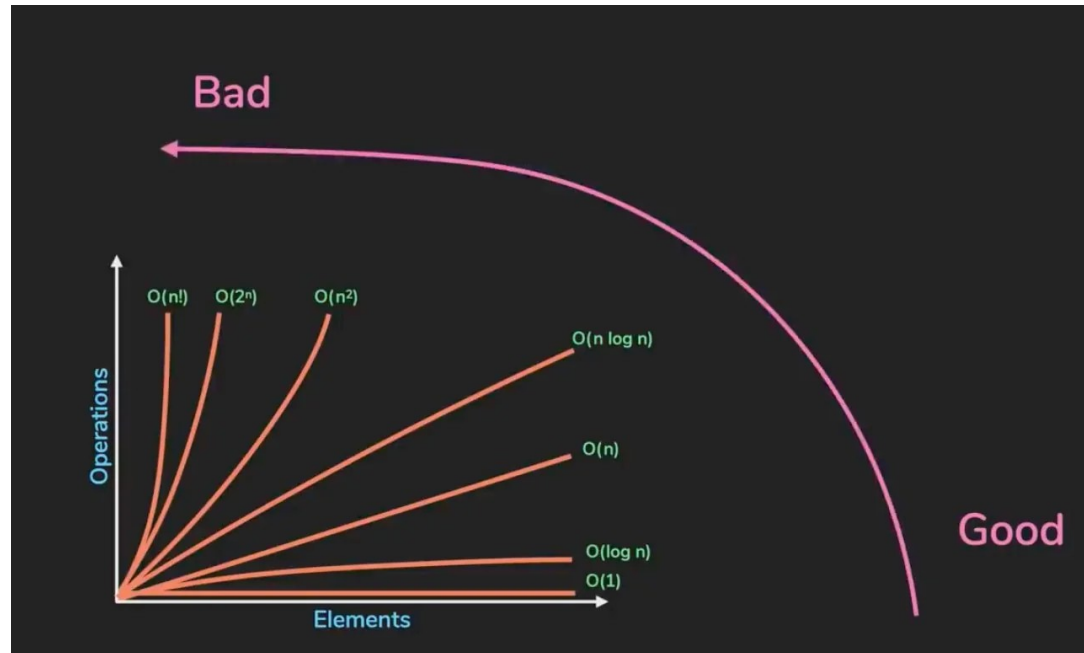


Сложност на алгоритъм

- Трябва ни оценка ама може да не е най-точната. Но пак да отговаря относително на реалността. Много е орязана, но все пак помага.
- Малко като на изпитите. Не ни пука 3 4 5 или 6, а ни пука минаваме или не. Тук е същото.
- Трябва да преценим горе долу колко операции извършва кода ни

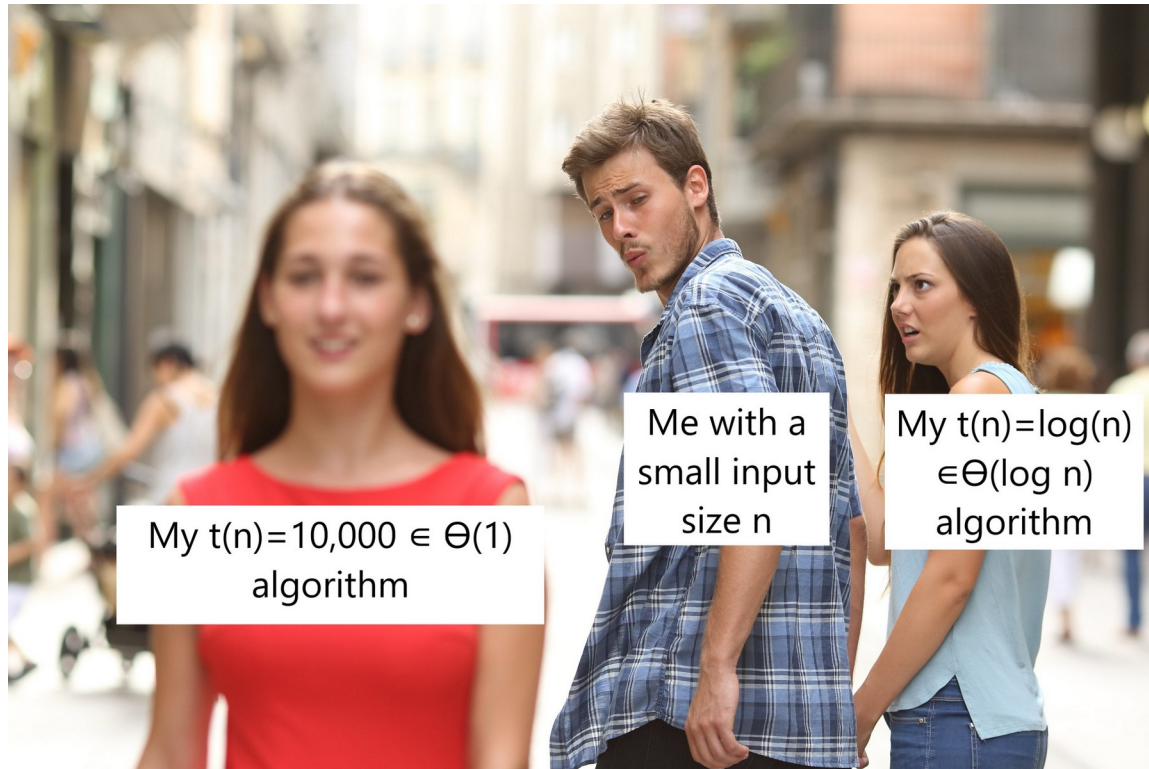
Big O Notation

- 3 класа
- Ние ще се занимаваме с O



Log-a няма основа?

- Не ни пука за константи!!!



Двоично търсене

- Сортираният масив е най-добрия ни приятел.
- Интуиция: Как ще търсим число в сортиран масив? Ще гледаме целия или ако видим някое по-голямо ще гледаме наляво от него?

Именно

- Тъй като е сортиран не е нужно винаги целия да се гледа. Гледаме средата. Ако търсеното е по-малко гледаме ляво. Ако не – дясно.
- Каква сложност има това?

Свети log

- Реално можем да го съобразим от тук
- На всяка итерация размерът намалява 2 пъти.
Реално с някакъв краен брой итерации означаваме го с k ще стане 0 или 1 тоест $2^k \geq n$.
Значи с k итерации сме приключили масива. Да но това k е $\log_2(n) + 1$. И имаме че това нещо е по-голямо от броя итерации следователно $O(\log(n))$.

Сложно, а?

- Реално не е чак толкова сложно. Преди предмета ДАА 2 курс 2 семестър ви е достатъчна само интуиция кое с каква сложност е. А след него също :). Реално ще развиете такава

Двоично търсене по отговора

- “Когато не се сещаш за нищо друго, пробвай дали ще стане с двоично търсене” ползотворен цитат от ПМГ-то.
- Същността е, че в този случай ние нямаме масив или нещо такова. Трябва да намерим отговор на задача и имаме начин бързо да проверим дали този отговор е коректен или е по-голям или по-малък от верния.

Стана манджа с грозде. Така и трябва :)

- Примерна задача от контрола за национален отбор(Щото не ни се занимава с глупости):

Задача СК6. ПЛОЧКИ

Автор: Кинка Кирилова-Лупанова

Кметът на град Видволп наел двама майстори - Димитър и Цветан, да поставят X нови плочки на градската спортна площадка.

Димитър полага по A плочки на ден, но всеки K -ти ден почива и не слага нито една плочка. По този начин Димитър почива на K -я, $2K$ -я, $3K$ -я ден, и т.н.

Цветан полага по B плочки на ден, но всеки M -ти ден почива и не слага нито една плочка. По този начин Цветан почива на M -я, $2M$ -я, $3M$ -я ден, и т.н.

Майсторите работят паралелно, и затова в дните, когато никой от тях не почива, те поставят $A + B$ плочки. В дните, когато почива само Цветан – се полагат A плочки, а в дните, когато почива Димитър – B плочки. В дните, когато и двамата майстори почиват – нито една плочка не се полага.

Кметът иска да знае за колко дни майсторите ще наредят всички плочки и кога той ще може да открие спортната площадка.

Напишете програма **tile**, която намира броя на дните, за които ще бъдат поставени всички плочки.

Вход

От първия ред на стандартния вход се въвеждат пет цели числа A , K , B , M и X , разделени с по един интервал.

Изход

На един ред на стандартния изход програмата трябва да изведе едно цяло число – търсения брой дни.

Ограничения

$$1 \leq A, B \leq 10^9$$

$$2 \leq K, M \leq 10^{18}$$

$$1 \leq X \leq 10^{18}$$

Пример

Вход

2 4 3 3 25

Изход

7



Време за още задачи

- Каква е сложността на следния фрагмент код:

```
for(int i = 0; i < m; i++)  
{  
    for(int j = 0; j < n; j++)  
    {  
        std::cout << 4;  
    }  
}
```

- Реализирайте двоично търсене в сортиран масив

Sysadmin

Дизайн и Анализ на Алгоритми, 2011

Time Limit: 0.2s, Memory Limit: 64MiB

Обир на банка. Един от маскираните обирджии говори по телефона:

- "Шефе, проникнахме в банката, взехме заложници, срязахме кабелите за тока, започнахме да разбиваме сейфа... обаче през вентилацията се появи брадясал мъж със захабени дрехи, обезвреди ни, взе ни резачките и оръжията..."

- "И какво, освободи заложниците?"

- "А, не, почна да лепи кабелите и да мърмори нещо за "uptime"."

- "Ясно. Имаме си работа със сисадмин..."

Работата на системния администратор изобщо не е толкова лесна. Обирджиите са нарязали кабелите на N парчета с дължини A_1, A_2, \dots, A_N . За да възстанови връзката на всички компютри, на него са му нужни K парчета кабел с еднаква дължина. Затова той иска да нареже намиращите му се под ръка N парчета на (евентуално) по-малки такива, така че да има K парчета кабел с еднаква дължина. Разбира се, колкото по-дълги са тези K парчета, толкова по-добре.

Вход

На първия ред на стандартния вход ще бъдат зададени целите числа N и K - съответно броя парчета, които са оставили обирджиите, и броя парчета с еднаква дължина, от които се нуждае сисадминът. На втория ще бъдат зададени N цели числа A_1, A_2, \dots, A_N , разделени с интервали - дължините на останените от обирджиите парчета.

Изход

На стандартния изход изведете един ред - максималната целочислена дължина, с която сисадминът може да изреже K кабела с еднаква дължина. Ако това е невъзможно дори при дължина едно, изведете 0.

Ограничения

- $1 \leq N \leq 100,000$
- $1 \leq K \leq 1,000,000,000$
- $1 \leq A_i \leq 1,000,000,000$

Примерен Вход	Примерен Изход
3 4 5 3 5	2
5 42 1 2 3 4 5	0
11 42 33 17 42 13 7 5 23 20 1 18 6	4

Load

Турнир за Купата на Декана, 2010

Time Limit: 0.2s, Memory Limit: 64MiB

Ели си избира кола! И за разлика от повечето други жени, които гледат предимно качества от сорта на това какъв цвят е тя, дали огледалото на сенника е достатъчно голямо за да си направят грима, хубав ли е звука в колата, климатикът и парното дали са достатъчно силни и т.н., Елеонора е по-практична. За нея най-важният фактор е резервоарът. Тя наистина не обича да спира да зарежда гориво (чупи си маникюра, после ръцете ѝ миришат на нафта, губи се време и т.н), от друга страна пък твърде голям резервоар излиза по-скъпо.

Ели е готова да направи компромис, като избере кола с най-малкият резервоар, достатъчно голям за да я закара с не повече от **K** зареждания от тях до училище. Пътя можем да си представим като права линия, върху която на позиция 0 се намира домът ѝ, на позиция **L** се намира ФМИ, а между тях има **N** бензиностанции на позиции **A₁**, **A₂**, ..., **A_N**. Можем да приемем, че колата харчи един литър гориво за единица разстояние.

Намерете минималната вместимост на резервоара, която би позволила на Ели да стигне от точка 0 до точка **L** с не повече от **K** зареждания. Приемаме, че в началото на пътуването резервоарът е пълен.

Вход

На първия ред на стандартния вход ще бъдат зададени целите числа **N**, **K** и **L**, съответно указващи броя бензиностанции, на които Ели може да зареди, максималният брой зареждания, които Ели може да направи, и дължината на пътя от тях до университета. На следващия ред ще има **N** цели числа **A₁**, **A₂**, ..., **A_N**, указващи позициите на бензиностанциите.

Изход

На единствен ред на стандартния изход изведете едно цяло число, указващо колко е минималната вместимост на резервоара, който отговаря на изискванията на Ели.

Ограничения

- $0 \leq K \leq 1,000$
- $1 \leq N \leq 100,000$
- $1 \leq L \leq 1,000,000,000$
- $0 < A_1 < A_2 < \dots < A_N < L$

Примерен Вход	Примерен Изход
2 1 42 13 23	23
8 2 1337 13 133 411 730 733 911 1113 1276	500

В първия пример Ели може да спре най-много веднъж на някоя от бензиностанциите в позиции 13 и 23. Университетът се намира в позиция 42. С резервоар с вместимост 23 тя може да стигне от тях до втората бензиностанция, да зареди там, и после да стигне до крайната си цел. Във втория пример тя може да зареди в бензиностанциите с позиции 411 и 911.

ЗАДАЧА Подредица

Дадена е редица от **N** числа. Пенчо се опитва да намери най-голямото число **L**, такова че в зададената редица съществува подредица от **K** на брой последователни числа всяко от което е по-голямо или равно на **L**. Тъй като Пенчо не е карал практикум по ДАА, ви моли да напишете програма която намира въпросното **L**.

Вход

1-ви ред: 2 числа, **N** и **K**, разделени с интервал ($1 \leq N \leq 1\,000\,000$, $1 \leq K \leq N$)

следващите **N** реда: числата от редицата, по едно на ред, всяко от което $\leq 2\,000\,000$.

Изход

Един ред с числото **L** – най-голямото число за което съществува желаната подредица с дължина **K**.

Пример

Вход	Изход
10 3 3 5 7 6 2 9 1 4 4 7	5

(Най-голямото възможно **L** е 5; при него имаме подредицата 5, 7 и 6)

Благодаря ви за вниманието