Searching and Divide-and-Conquer

Divide-and-conquer In the divide-and-conquer method, we divide a problem into several subproblems (of constant fraction size), solve each subproblem recursively, and combine the solutions to the subproblems to arrive at the solution to the problem.

To be efficient, it is important to balance the sizes of the subproblems.

Searching Problems are usually stated in the form of searching for an object. When you are searching for an object, it may be useful to explicitly identify the possible scenarios. A typical algorithm for a search problem consists of a series of steps in which we perform computation and ask a question to narrow down the possibilities. Each possible answer to the question reduces the problem to a subproblem where we have fewer possibilities for the object we are seeking. Moreover, the set of possible scenarios is partitioned along the answers to the question. For efficiency (that is, for minimizing the number of questions in the worst-case), it is important to design the algorithm so that the number of possibilities in each case is as equal as it can be to the number of possibilities in other cases.

1 Problems

Problem 1: A Fake among Eight Coins

There are eight identical-looking coins; one of these coins is counterfeit and is known to be lighter than the genuine coins. What is the minimum number of weighings needed to identify the fake coin with a two-pan balance scale without weights?

Solution. Divide the coins into three groups of 3, 3, and 2 coins each and call them L, R, and E respectively. Use the two-pan balance to compare the weights of the groups L and R. If they weigh the same, the fake coin can neither be in L nor in R, so it must be in E. Take the two coins in E and compare them. The lighter coin (between the two coins in E) must be the fake coin.

If the weights of L and R are different, then the fake coin must be in the group that is lighter. Take any 2 coins from this group and weigh them. If they weigh the same then the remaining coin in the group must be fake. If one of them is lighter, the lighter coin must be the fake coin.

Generalization

Suppose, instead of eight coins, there are now $n \geq 1$ coins. As before all the coins are equal in weight, except for one, which is lighter. We will argue that a general version of the previous algorithm would find the lighter coin in at most $\lceil \log_3 n \rceil$ comparisons. More precisely, we show that our algorithm performs exactly $\lceil \log_3 n \rceil$ comparisons in the worst-case.

Algorithm **SearchForLighterCoin**: Without loss of generality, let n be equal to $3k$ or $3k + 1$ or $3k + 2$ for some $k \geq 0$. If $n = 3k$ or $3k + 1$, form two groups of k coins each. Otherwise (that is, if $n = 3k + 2$), form two groups of $k + 1$ coins each. In either case, place the remaining coins in the third group. Call these groups L, R, and E respectively. All groups will have at least one coin as long as $k \geq 1$.

Compare groups L and R. If they weigh the same, the lighter coin must be in E. If the weights of L and R are different, the lighter coin must be in the lighter group. Consider the group with the lighter coin and apply the procedure recursively until the number of coins is 1 or 2, at which point lighter coin can be detected with at most one comparison.

Analysis: The analysis is a little tricky. It turns out that it is easier to work with the ternary representation of n since the divide-and-conquer scheme creates a subproblem of size about $n/3$.

Without loss of generality, let $3^k \leq n < 3^{k+1}$ for some $k \geq 0$. Write the ternary representation of n without the leading zeros. At a risk of slight ambiguity, we will use n to denote its ternary representation as well. The length of the ternary representation of n, $|n|_3$, is the number of trits (for ternary digits) where we count the trits from the least significant trit to the leading trit (the leftmost non-zero trit). Observe that $\lceil \log_3 n \rceil = k + 1$ if $n > 3^k$, and $\lceil \log_3 n \rceil = k$ if $n = 3^k$. Our analysis proceeds along these two cases. In each case, we will show that the algorithm performs at most $\lceil \log_3 n \rceil$ comparisons.

We first consider the case where $n = 3^k$ and show that the algorithm performs exactly k comparisons to find the lighter coin by induction on k. If $n = 3^k$, then a comparison of groups L and R reduces the problem to that of size exactly $n/3 = 3^{k-1}$. Thus, by induction, we argue that it takes $k - 1$ more comparisons for a total of $k = \lceil \log_3 n \rceil$ comparisons to find the lighter coin. We do not need to perform any comparisons to figure out the lighter coin when $k = 0$ (base case for the induction).

We will now consider the case $3^k < n < 3^{k+1}$ and argue that the algorithm performs at most $k + 1$ comparisons by induction on k. If $k = 0$, then $n = 2$ in which case the algorithm performs exactly one comparison.

If $k \geq 1$, write the ternary representation of n as $n't$ where $n' \geq 1$ is a number in ternary representation and t a single trit. After one comparison, we end up with a problem with at most $n' + 1$ coins. Since $3^k < n < 3^{k+1}$, we get that $3^{k-1} < n' + 1 \leq 3^k$. If $n' + 1 = 3^k$, from the earlier analysis tells us that the algorithm performs exactly k more comparisons to solve the problem. Otherwise, by induction, we argue that the algorithm performs at most k more comparisons. In total, the algorithm performs at most $k + 1 = \lceil \log_3 n \rceil$ comparisons as needs to be shown.
Notes: For all $n \geq 1$, the algorithm performs exactly $\lceil \log_3 n \rceil$ comparisons in the worst case. In fact, for all $n \geq 1$, the number of comparisons in the best and the worst case differ by at most 1. To see this, observe that each comparison (except for the last comparison) could yield a subproblem of size exactly $\lceil n/3 \rceil$.

Variants

If there are two fake coins instead of one, we need 6 comparisons. Divide the 8 coins into 4 pairs and compare each pair. If any coin is lighter, there must be two coins which are lighter. These two coins are the lighter coins. If all comparisons yield equality, the fake coins must be together in the same pair. Take one coin from each of the four pairs. Use the previous algorithm to determine the lighter coin using two more comparisons.

History

References

The problems considered in this section belong to the more general class of problems, called group testing.

References

[1] D. Du and F. Hwang. *Combinatorial Group Testing and Its Applications*. Applied Mathematics. World Scientific, 2000.

[2] Piotr Indyk, Hung Q. Ngo, and Atri Rudra. Efficiently decodable non-adaptive group testing. In *Proceedings of the Twenty-first Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '10, pages 1126–1142, Philadelphia, PA, USA, 2010. Society for Industrial and Applied Mathematics.

Problem 2: Lighter or Heavier

You have $n > 2$ identical-looking coins and a two-pan balance scale with no weights. One of the coins is a fake, but you do not know whether it is lighter or heavier than the genuine coins, which all weigh the same. Design an algorithm to determine in the minimum number of weighings whether the fake coin is lighter or heavier than the others.

Solution.

A suboptimal solution: Create 3 groups of $\lfloor \frac{n}{3} \rfloor$ coins each. Call these groups A, B, and C. The remaining coins form the group E. E contains 0, 1, or 2 coins. Compare groups A and B and groups A and C. Comparison of any two groups of coins using a two-pan balance would produce one of three outcomes: the first group is lighter, the first group is heavier, or both groups have equal weight. The information obtained as a result of the two comparisons is best represented as a 3-way tree with $3 \times 3 = 9$ leaves. Each leaf corresponds to a set of possibilities. The following table captures the two-level 3-way tree. We use the notation $A < B$ to denote that the weight of coins in group A is less than the weight of the coins in group B. $A > B$ and $A = B$ are used with an analogous meaning.

A vs B	A vs C	Analysis
$A < B$	$A < C$	$B = C$. The fake coin is lighter and is in A
	$A = C$	$A = C$. The fake coin is heavier and is in B
	$A > C$	A, B, and C have distinct weights, which is impossible
$A = B$	$A < C$	$A = B$. The fake coin is heavier and is in C
	$A = C$	$A = B = C$. The fake coin is in E
	$A > C$	$A = B$. The fake coin is lighter and is in C
$A > B$	$A < C$	A, B, and C have distinct weights, which is impossible
	$A = C$	$A = C$. The fake coin is lighter and is n B
	$A > C$	$B = C$. The fake coin is heavier and is in A

For row 5, we concluded that all the coins in the groups A, B, and C are genuine and the fake coin is in the group E. To figure out whether the fake coin is lighter or heavier, take $|E|$ genuine coins from among the

coins in A, B, and C. Compare this group with E to determine whether the fake coin is lighter or heavier. In total, we need a maximum of 3 comparisons.

An optimal solution: It turns out that we can devise an algorithm for this problem that uses just two comparisons.

Algorithm **DetermineLighterOrHeavier**: Let n be equal to $3k$ or $3k+1$ or $3k+2$ for some $k \geq 1$. This is without loss of generality since $n > 2$. If $n = 3k$ or $3k + 1$, form two groups of k coins each. Otherwise (that is, if $n = 3k + 2$), form two groups of $k + 1$ coins each. In either case, place the remaining coins in the third group. Call these groups L, R, and E respectively. All groups have at least one coin since $k \geq 1$. We use the notation $|\cdot|$ to denote the size of a group. For example, $|L|$ denotes the number of coins in L.

Compare the weights of groups L and R. If that the weights are equal (denoted by $L = R$), we know then that each coin in groups L and R is genuine. Select $|E|$ many coins from the groups L and R and form a new group E'. Compare the weights of the groups E and E'. If $E < E'$, then the fake coin is lighter. If $E > E'$, the fake coin is heavier. It is not possible that $E = E'$. Observe that $|L| + |R| \geq |E|$ for all $n > 2$, so it is always possible to select $|E|$ many coins from the groups L and R together.

Now consider the case $L < R$. We know that each coin in E is genuine. If $|L|$ is odd, add a genuine coin from E to L. Without loss of generality, assume that $|L| \geq 2$ is even. Split L into two subgroups of equal size. Call the subgroups L' and L''. Compare L' and L''. If $L' = L''$, we infer that the fake coin is heavier since all the coins in L are genuine. If $L' < L''$ or $L' > L''$, we infer that the fake coin is lighter. since otherwise we will have

The case $L > R$ can be handled similarly.

Thus, in call cases, we have shown that we need only two comparisons to determine whether the fake coin is lighter or heavier.

Problem 3: Twelve Coins

There are 12 coins identical in appearance; either all are genuine or exactly one of them is fake. It is unknown whether the fake coin is lighter or heavier than the genuine one. You have a two-pan balance scale without weights. The problem is to find whether all the coins are genuine and, if not, to find the fake coin and establish whether it is lighter or heavier than the genuine ones. Design an algorithm to solve the problem in the minimum number of weighings.

Solution.

A suboptimal solution: The problem can be solved using the techniques used in solving the problems **A Fake among 8 coins** and **Lighter or Heavier**. Divide the 12 coins into 3 groups of 4 coins each. Call the groups A, B, and C. Compare A and B as well as A and C. We will use the analysis from the solution to the problem **Lighter of Heavier** figure out whether the fake coin is lighter or heavier and which group it belongs to. Remember the group E has zero coins. In all the possible cases except for case 5, the relative weight of the fake coin and the group it belongs are identified. With two additional comparisons, we figure out fake coin itself using the algorithm for the problem **Fake Coin**. For case 5, we conclude that all the coins are genuine, since the group E has no coins in it,

Overall, we need a maximum of 4 comparisons to figure out the fake coin.

An optimal solution: Let $c_1, c_2, \ldots, c_{12}$ denote the coins. Let $L = \{c_1, c_2, c_3, c_4\}$, $R = \{c_5, c_6, c_7, c_8\}$, and $E = \{c_9, c_{10}, c_{11}, c_{12}\}$. Compare L and R. We discuss each of the three cases below.

$L = R$

In this case, if there is a fake coin, it must be in E. We also know that all the coins in L and R are genuine. Let $L' = \{c_1, c_9\}$, $R' = \{c_{10}, c_{11}\}$ and $E' = \{C_{12}\}$. Compare L' and R'.

If $L' = R'$, then the only possibilities are either all the coins are genuine or c_{12} is a fake (that is, it is either lighter or heavier). By comparing c_{12} with a genuine coin, we can determine whether it is genuine, lighter or heavier.

If $L' < R'$, we then have three possibilities since one of c_9, c_10, and c_11 must be a fake coin.

- c_9 is lighter, and c_{10}, c_{11}, and c_{12} are genuine

- c_{10} is heavier, and c_9, c_{11} and c_{12} are genuine

- c_{11} is heavier, c_9, c_{10} and c_{12} are genuine

We compare c_{10} and c_{11} to determine which coin is fake and whether it is lighter or heavier.

If $L' > R'$, we once again have three possibilities since one of c_9, c_10, and c_11 must be fake. We compare c_{10} and c_{11} to determine which coin is fake and whether it is lighter or heavier.

$L < R$

In this case, we know that the coins in E are genuine and either one of the coins in L is lighter or one of the coins in R is heavier. Altogether we have 8 possibilities.

Let $L' = \{c_9, c_6, c_4\}$, $R' = \{c_3, c_7, c_8\}$, and $E' = \{c_1, c_2, c_5\}$. Compare L' and R'.

If $L' = R'$, then we have three possibilities: c_1 is lighter, c_2 is lighter, or c_5 is heavier. By comparing c_1 and c_2, we can resolve the remaining uncertainty.

If $L' < R'$, we again have three possibilities: c_4 is lighter, c_7 is heavier, or c_8 is heavier. By comparing c_7 and c_8, we can find the fake coin and its relative weight.

If $L' > R'$, we have only two possibilities: c_6 is heavier or c_3 is lighter. A comparison between c_6 and c_7 would eliminate the remaining uncertainty.

$L > R$

This case can be handled similar to the previous case.

In conclusion, we have shown that we need three comparisons to determine whether there is an exceptional coin, and if so which coin it is and whether it is lighter or heavier.

Problem 4: A Stack of Fake Coins
There are 10 stacks of 10 identical-looking coins. All of the coins in one of these stacks are counterfeit, and all the coins in the other stacks are genuine. Every genuine coin weighs 10 grams, and every fake weighs 11 grams. You have an analytical scale that can determine the exact weight of any number of coins. What is the minimum number of weighings needed to identify the stack with the fake coins?

Solution. One weighing is sufficient. Let $S_1, S_2, \ldots, S_{10}$ denote the ten stacks. Create a group of coins by selecting i coins from stack S_i for every $1 \leq i \leq 10$. Use the scale to determine their weight. If the weight is w, the counterfeit stack is then S_{w-550}.

There are $\frac{(10)(10+1)}{2} = 55$ coins in the group. Each genuine coin weighs 10 grams, so the entire group, would weigh 550 grams if every S_i is a stack of genuine coins. However, there are between 1 and 10 fake coins in the group depending on which stack contains the fake coins. For any $1 \leq i \leq 12$, if S_i is the stack of fake coins, the weight of the group would be $550 + i$. Hence, subtracting 550 from the weight of the group should give us the index of the stack of fake coins.

Problem 5: Max-Min Weights

Given $n > 1$ items and a two-pan balance scale with no weights, determine the lightest and the heaviest items in $\lceil 3n/2 \rceil - 2$ weighings.

Solution. Divide the objects into pairs. Set aside the unpaired object, if it exists. Compare the objects in each pair. Add the lighter object to group A and the heavier object to group B. If there is a tie, add the one of the objects to group A and the other to group B. Add the unpaired object to both groups.

Observe that group A contains the lightest object and group B contains the heaviest object. Use the following procedure to find the lightest object from A.

Assume that the objects in A are arranged in a sequence. Scan the objects in A in order while comparing the next object to the lightest object found so far. Initially, the lightest object found so far is the first object in the sequence. After all objects in A are scanned, lightest object found so far will be the lightest object in A.

We find the heaviest object in B by using a similar procedure.

There are $\lfloor \frac{n}{2} \rfloor$ comparisons in the first phase to determine the groups A and B. Finding the lightest object from group A requires exactly $\lceil \frac{n}{2} \rceil - 1$ comparisons. Similarly, it takes $\lceil \frac{n}{2} \rceil - 1$ comparison to find the heaviest object from group B. In total, we need exactly $\lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{2} \rceil - 1 + \lceil \frac{n}{2} \rceil - 1 = n + \lceil \frac{n}{2} \rceil - 2 = \lceil \frac{3n}{2} \rceil - 2$ comparisons.

2 Advanced Problems