Introduction to Algorithms

Ulises (Tirado Zatarain | Mendez Martinez)

Apr, 2021

Two element Boolean algebra

It has only two elements, **0** and **1**, and is defined by the rules:

$$\begin{bmatrix} \wedge & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \vee & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a & 0 & 1 \\ \neg a & 1 & 0 \end{bmatrix}$$

It is also used for circuit design in electrical engineering; here 0 and 1 represent the two different states of one bit in a digital circuit, typically high and low voltage.

Deduction

In deductive reasoning (*"top-down logic"*), a conclusion is reached reductively by applying general rules which hold over the entirety of a closed domain of discourse, narrowing the range under consideration until only the conclusion(s) remains. In deductive reasoning there is no epistemic uncertainty.

- The task is to find how many numbers less than or equal to N have numbers of divisors exactly equal to 3.

Mathematical induction

- Mathematical proof technique.
- Used to prove that a statement $P(n)$ holds for every natural number $n = 0, 1, 2, 3, \ldots$;
- The mathematical method examines infinitely many cases to prove a general statement, but does so by a finite chain of **deductive reasoning** involving the variable n, which can take infinitely many values.
- Two cases:
  - **Base case** (or basis), proves the statement for an initial value (typically $n = 0$ or $n = 1$).
  - **Induction step**, proves that if the statement holds for any given case $n = k$, then it must also hold for the next case $n = k + 1$.

- Mathematical proof technique.
- Used to prove that a statement $P(n)$ holds for every natural number $\$n = 0, 1, 2, 3, \ldots\$$;
- The mathematical method examines infinitely many cases to prove a general statement, but does so by a finite chain of **deductive reasoning** involving the variable n, which can take infinitely many values.
- Two cases:
  - **Base case** (or basis), proves the statement for an initial value (typically $n = 0$ or $n = 1$).
  - **Induction step**, proves that if the statement holds for any given case $n = k$, then it must also hold for the next case $n = k + 1$.

Mathematical induction

- Mathematical proof technique.
- Used to prove that a statement $P(n)$ holds for every natural number $n = 0, 1, 2, 3, \ldots$;
- The mathematical method examines infinitely many cases to prove a general statement, but does so by a finite chain of **deductive reasoning** involving the variable n, which can take infinitely many values.
- Two cases:
  - **Base case** (or basis), proves the statement for an initial value (typically $n = 0$ or $n = 1$).
  - **Induction step**, proves that if the statement holds for any given case $n = k$, then it must also hold for the next case $n = k + 1$.

Mathematical induction

- Mathematical proof technique.
- Used to prove that a statement $P(n)$ holds for every natural number \$n = 0, 1, 2, 3, . . . \$;
- The mathematical method examines infinitely many cases to prove a general statement, but does so by a finite chain of **deductive reasoning** involving the variable n, which can take infinitely many values.
- Two cases:
  - **Base case** (or basis), proves the statement for an initial value (typically $n = 0$ or $n = 1$).
  - **Induction step**, proves that if the statement holds for any given case $n = k$, then it must also hold for the next case $n = k + 1$.

Mathematical induction

- Mathematical proof technique.
- Used to prove that a statement $P(n)$ holds for every natural number \$n = 0, 1, 2, 3, . . . \$;
- The mathematical method examines infinitely many cases to prove a general statement, but does so by a finite chain of **deductive reasoning** involving the variable n, which can take infinitely many values.
- Two cases:
  - **Base case** (or basis), proves the statement for an initial value (typically $n = 0$ or $n = 1$).
  - **Induction step**, proves that if the statement holds for any given case $n = k$, then it must also hold for the next case $n = k + 1$.

Mathematical induction

- Mathematical proof technique.
- Used to prove that a statement $P(n)$ holds for every natural number $n = 0, 1, 2, 3, \ldots$;
- The mathematical method examines infinitely many cases to prove a general statement, but does so by a finite chain of **deductive reasoning** involving the variable n, which can take infinitely many values.
- Two cases:
  - **Base case** (or basis), proves the statement for an initial value (typically $n = 0$ or $n = 1$).
  - **Induction step**, proves that if the statement holds for any given case $n = k$, then it must also hold for the next case $n = k + 1$.

Triangular numbers

- A **triangular number** or **triangle number** counts objects arranged in an equilateral triangle.
- The nth triangular number is the number of dots in the triangular arrangement with n dots on a side, and is equal to the sum of the n natural numbers from 1 to n.
- \$0, 1, 3, 6, 10, 15, 21, 28, 36, 45, 55, . . .\$
- TBD: Agregar imagen

Triangular numbers

- A **triangular number** or **triangle number** counts objects arranged in an equilateral triangle.
- The nth triangular number is the number of dots in the triangular arrangement with n dots on a side, and is equal to the sum of the n natural numbers from 1 to n.
- $0, 1, 3, 6, 10, 15, 21, 28, 36, 45, 55, \ldots$
- TBD: Agregar imagen

Triangular numbers

- A **triangular number** or **triangle number** counts objects arranged in an equilateral triangle.
- The nth triangular number is the number of dots in the triangular arrangement with n dots on a side, and is equal to the sum of the n natural numbers from 1 to n.
- $0, 1, 3, 6, 10, 15, 21, 28, 36, 45, 55, ...$
- TBD: Agregar imagen

Triangular numbers

- A **triangular number** or **triangle number** counts objects arranged in an equilateral triangle.
- The nth triangular number is the number of dots in the triangular arrangement with n dots on a side, and is equal to the sum of the n natural numbers from 1 to n.
- $0, 1, 3, 6, 10, 15, 21, 28, 36, 45, 55, . . . $
- TBD: Agregar imagen

Recursion

- TBD: Caso/Condición base.
- TBD: Relación de recurrencia.
- TBD: Stack overflow.

- TBD: Caso/Condición base.
- TBD: Relación de recurrencia.
- TBD: Stack overflow.

Recursion

- TBD: Caso/Condición base.
- TBD: Relación de recurrencia.
- TBD: Stack overflow.

Factorial

Solution I

Problem: https://leetcode.com/problems/convert-binary-number-in-a-linked-list-to-integer/

```
void process(ListNode* head, int &n) {
  if (head) {
    n <<= 1;
    n += head->val;
    process(head->next, n);
  }
}
```

Solution II

Problem: https://leetcode.com/problems/convert-binary-number-in-a-linked-list-to-integer/

TBD

References:

- https://en.wikipedia.org/wiki/Boolean_algebra_(structure)
- https://practice.geeksforgeeks.org/problems/3-divisors3942/1
- https://www.khanacademy.org/math/algebra-home/alg-series-and-induction
- https://en.wikipedia.org/wiki/Mathematical_induction
- https://en.wikipedia.org/wiki/Triangular_number
- https://oeis.org/A000217
- https://www.geeksforgeeks.org/recursion/
- https://www.khanacademy.org/computing/computer-science/algorithms#recursive-algorithms
- https://web.mit.edu/6.005/www/fa15/classes/10-recursion/#

Q & A