

Brute Force

We are going to give a name for something that we have been doing in the past assignments.

The way in which we are solving some problems comes under this category.

Brute Force

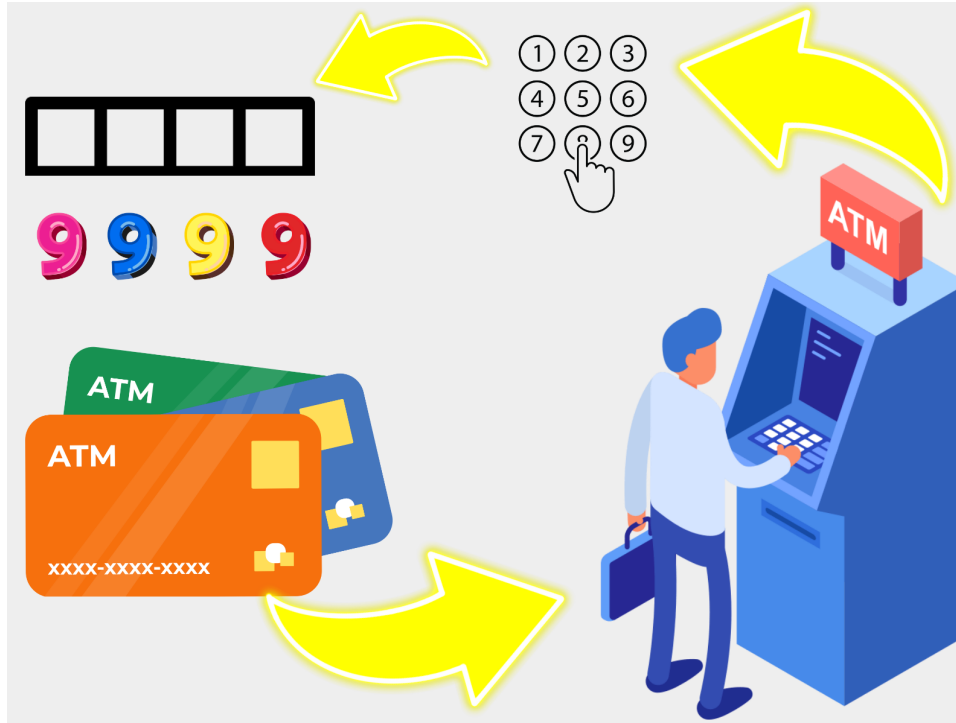
“Try all possible solutions (systematically)”

- A brute force approach is an approach that finds all the possible solutions to find a satisfactory solution to a given problem.
- The brute force algorithm tries out all the possibilities till a satisfactory solution is not found.

Example—

- **ATM ⇒ PIN** — There is some combination of numbers from 0-9 that will help you to withdraw the money.

Some combinations are— 0 0 0 0, 0 0 0 1, 0 0 0 2, ... and so on.



that's why banks give only 3 attempts to enter the correct pin.

- In Cryptography, there are **brute-force attacks**— In cryptography, some data is locked, and to unlock it you need a key.

Features of Brute-Force Algorithms

- **Naive /Blind Search**— No strategy involved.
- **Guaranteed to find the solution**— or report if no solution exists (guaranteed to be correct).
- **Often Easy to understand and implement**— (Yeh approach to definitely work Karega).
- **Often suffer from a “Combinatorial Explosion”**— the number of possible solutions increases extremely rapidly as the input size increases.

High-Level Description of Brute Force Algorithms

- Generate all possible candidate solutions.
- For each candidate's solution,

- check if it satisfies the constraints of the problem.
- pick the 'best' solution. (if needed) [optional]

Examples

Code 1:- Find all divisors of a given number `n`

Brute-Force Algorithm

- Generate all possible factors of n.
- 1,2,3,4,5,6,7,8,9,.....n-1,n
- for each possible "i", check if "i" divides n

```
# given number = n
n=18
for i in range(1,n+1):
    if(n%i==0):
        print(i)
```

Code 2:- Finding the longest-increasing sublist

Brute-Force Algorithm

- Generate all possible sublists.
- For each sub-list :
 - Check if it is increasing.
 - Compare its length with "max-length".
 - Update the "max-length" if necessary.

Happy Coding!