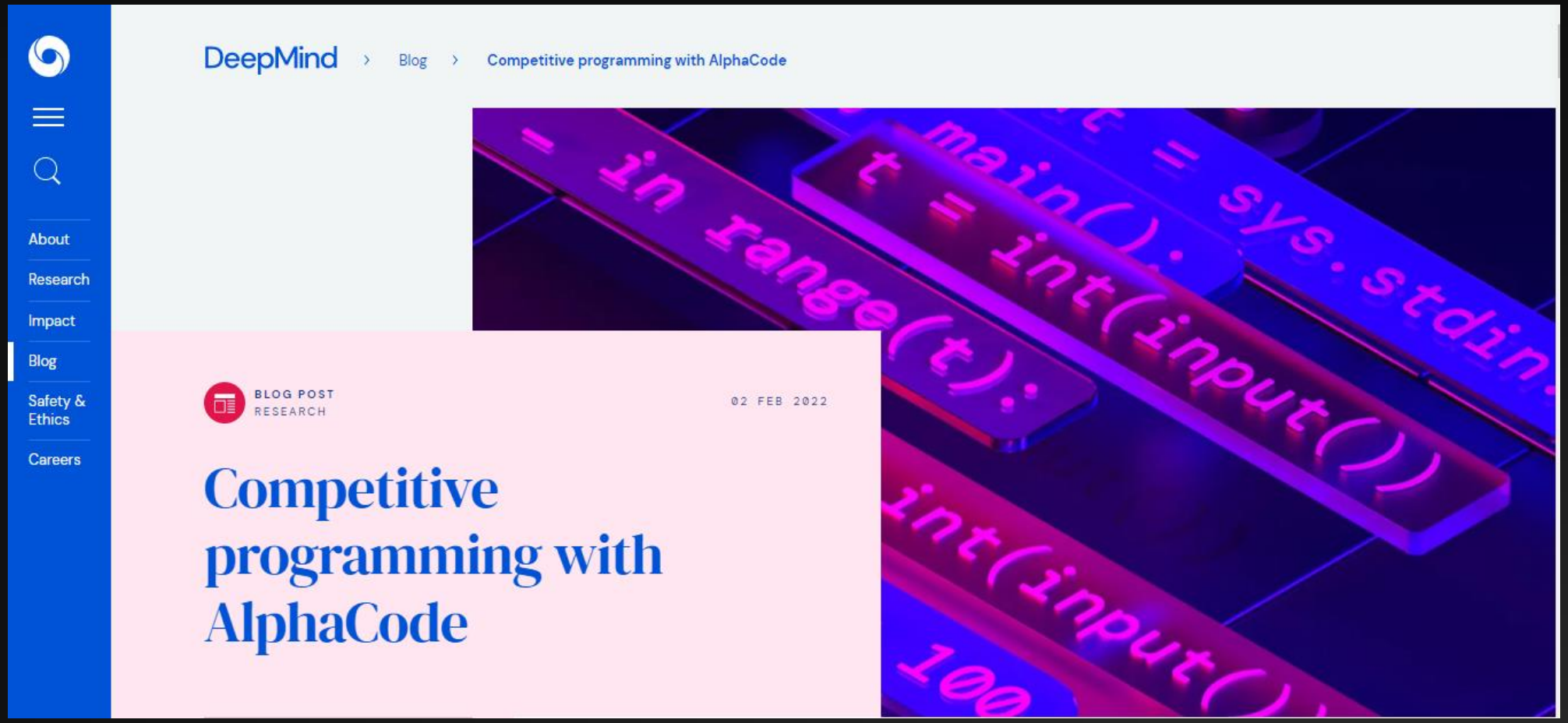
Lectrue_1

Intro to computer science

Provided by PBSCI

Welcome to computer science world



Alphacode by DeepMind

Problem description input

1

Problem (input)

D.Backspace

You are given two strings s and t, both consisting of lowercase English letters. You are going to type the string s character by character, from the first character to the last one.

When typing a character, instead of pressing the button corresponding to it, you can press the "Backspace" button. It deletes the last character you have typed among those that aren't deleted yet (or does nothing if there are no characters in the current string). For example, if s is "abcbd" and you press Backspace instead of typing the first and the fourth characters, you will get the string "bd" (the first press of Backspace deletes no character, and the second press deletes the character 'c'). Another example, if s is "abcaa" and you press Backspace instead of the last two letters, then the resulting text is "a".

Your task is to determine whether you can obtain the string t, if you type the string s and press "Backspace" instead of typing several (maybe zero) characters of s.

Input

The first line contains a single integer q ($1 \le q \le 10^5$) — the number of test cases.

The first line of each test case contains the string s ($1 \le |s| \le 10^5$). Each character of s is a lowercase English letter.

The second line of each test case contains the string t ($1 \le |t| \le 10^5$). Each character of t is a lowercase English letter.

It is guaranteed that the total number of characters in the strings over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, print "YES" if you can obtain the string t by typing the string s and replacing some characters with presses of "Backspace" button, or "NO" if you cannot.

You may print each letter in any case (YES, yes, Yes will all be recognized as positive answer, NO, no and nO will all be recognized as negative answer).

Input
4 ababa ba ababa bb aaa aaaa aababa ababa

Output
YES NO NO YES

Note

Consider the example test from the statement.

In order to obtain "ba" from "ababa", you may press Backspace instead of typing the first and the fourth characters.

There's no way to obtain "bb" while typing "ababa".

There's no way to obtain "aaaa" while typing "aaa".

In order to obtain "ababa" while typing "aababa", you have to press Backspace instead of typing the first character, then type all the remaining characters.

Source code output

```
t=int(input())
for i in range(t):
    s=input()
    t=input()
    a=[]
    b=[]
    for j in s:
        a.append(j)
    for j in t:
        b.append(j)
    a.reverse()
    b.reverse()
    c=[]
    while len(b)!=0 and len(a)!=0:
        if a[0]==b[0]:
            c.append(b.pop(0))
            a.pop(0)
        elif a[0]!=b[0] and len(a)!=1:
            a.pop(0)
            a.pop(0)
        elif a[0]!=b[0] and len(a)==1:
            a.pop(0)
    if len(b)==0:
        print("YES")
    else:
        print("NO")
```

First AlphaCode reads the two phrases.

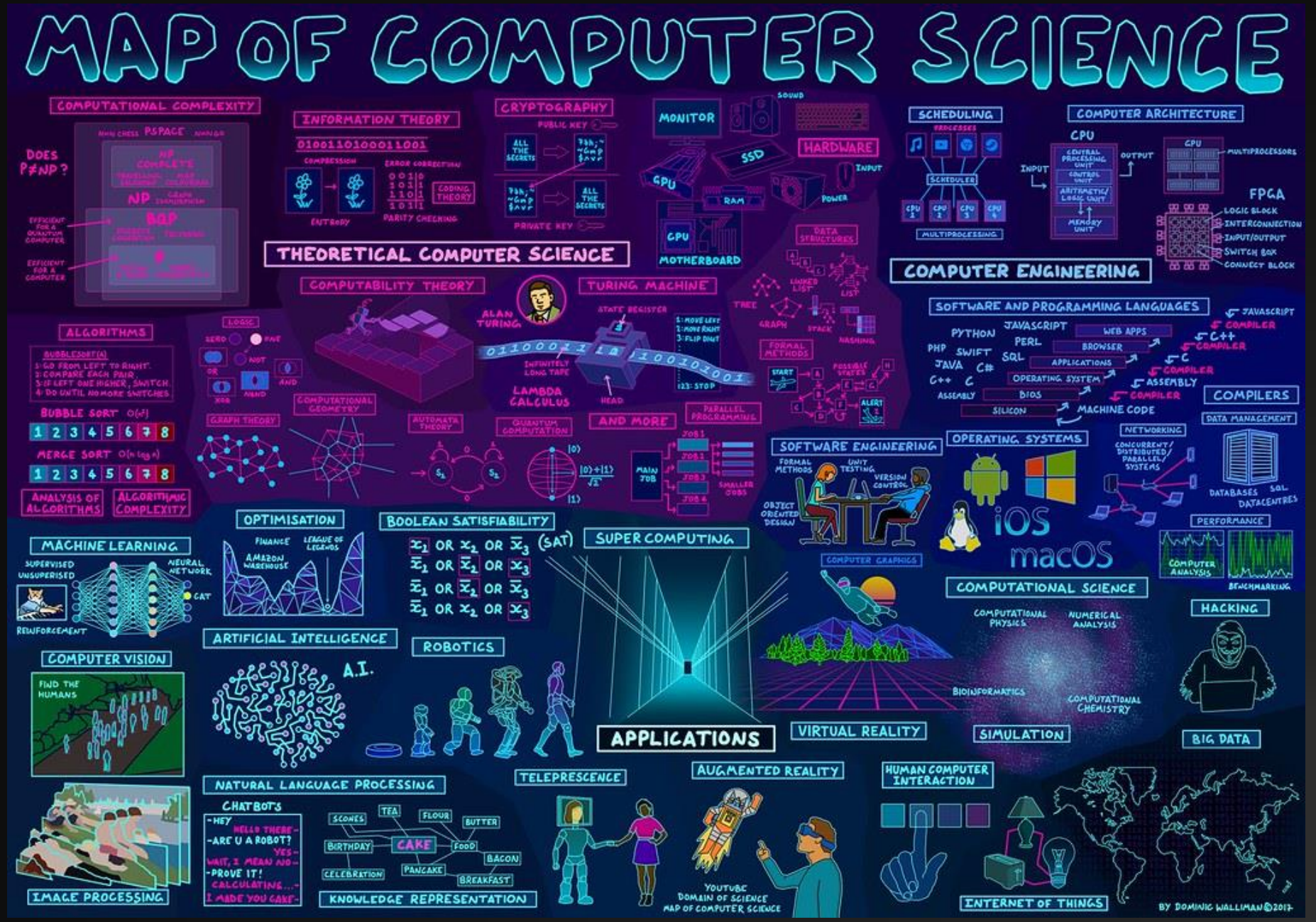
Backspace deletes two letters. The letter you press backspace instead of, and the letter before it.

If the letters at the end of both phrases don't match, the last letter must be deleted. If they do match we can move onto the second last letter and repeat.

If we've matched every letter, it's possible and we output that.

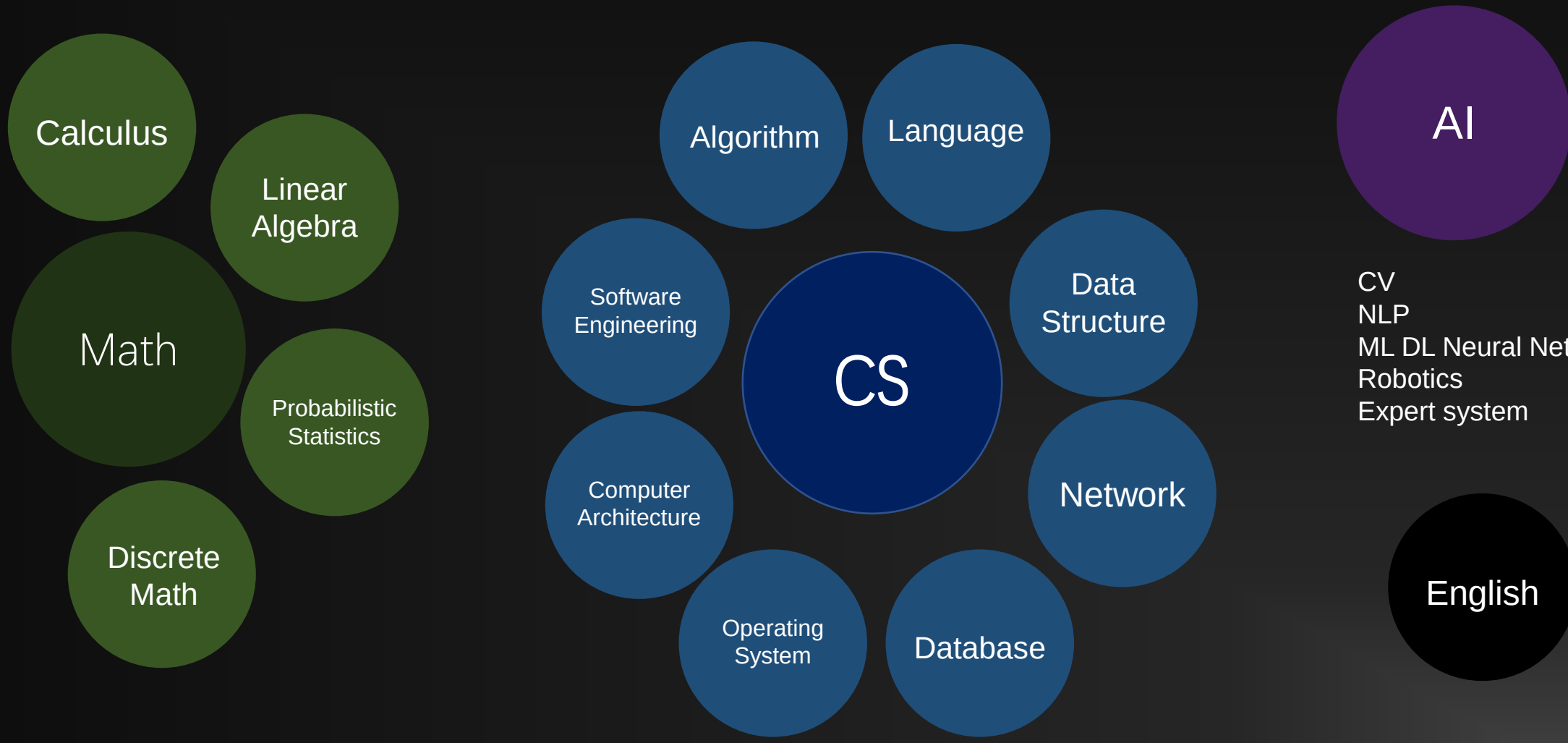
~~Programmer~~

Engineer



Map of Computer Science

source :
Domain of Science


Calculus
Linear Algebra
Math
Probabilistic Statistics
Discrete Math
Algorithm
Language
Software Engineering
CS
Data Structure
Computer Architecture
Network
Operating System
Database
AI
CV
NLP
ML DL Neural Networks
Robotics
Expert system
English

HTML

markup language

Bash

command-line interface language

Python

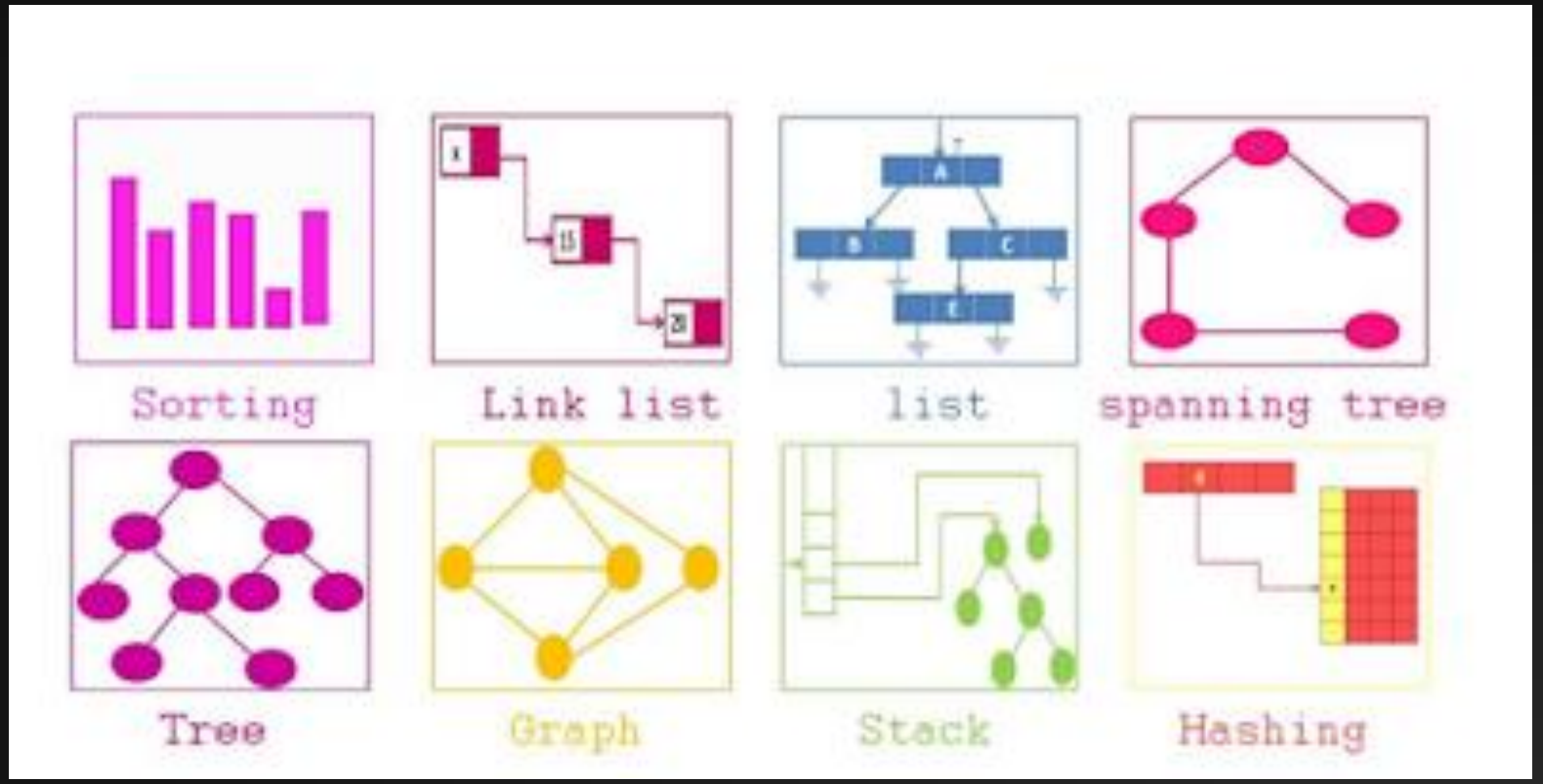
scripting language

C++

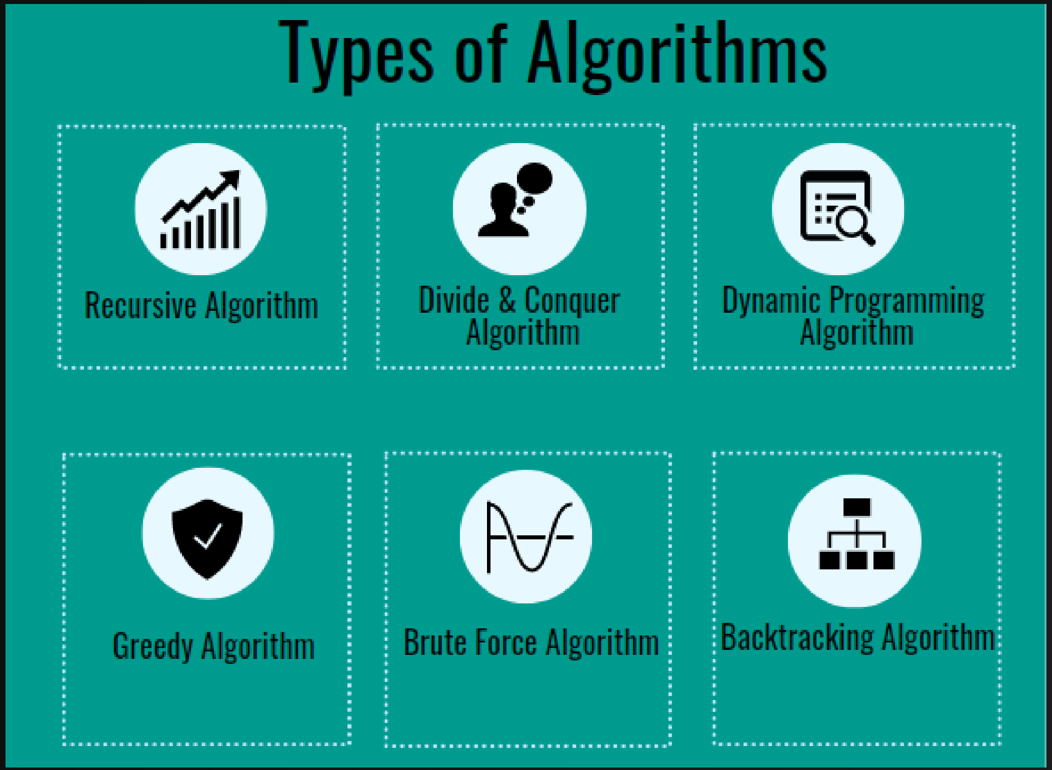
compiled language

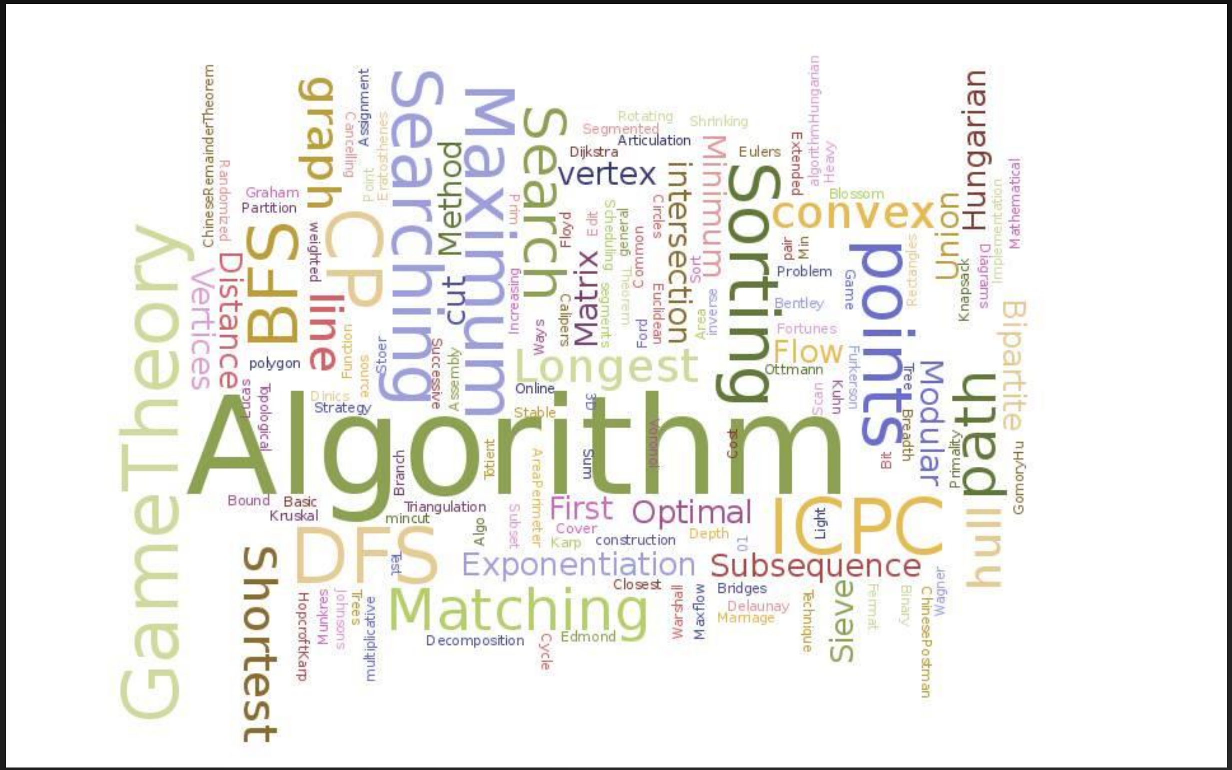
Programming Language


OpenEdge ABL
ABAP
Visual Basic
Transact-SQL
Clarion
FoxPro
Perl
Logo
Prolog
Objective-C
ColdFusion
Erlang
C#
D
Scala
F#
ML
JavaScript
Python
PostScript
Forth
Q
C++
ActionScript
PL/SQL
COBOL
Java
Assembly
Scratch
Scheme
R
Haskell
Go
Fortran
MATLAB
C
Ada
Lua
Io
Yacc
TCL
PHP
Delphi
PL/I
Ruby
Mathematica
Pascal
Dart
Scala
Max/MSP
cT
SAS
Haskell
Tcl
Groovy
Lisp
Swift

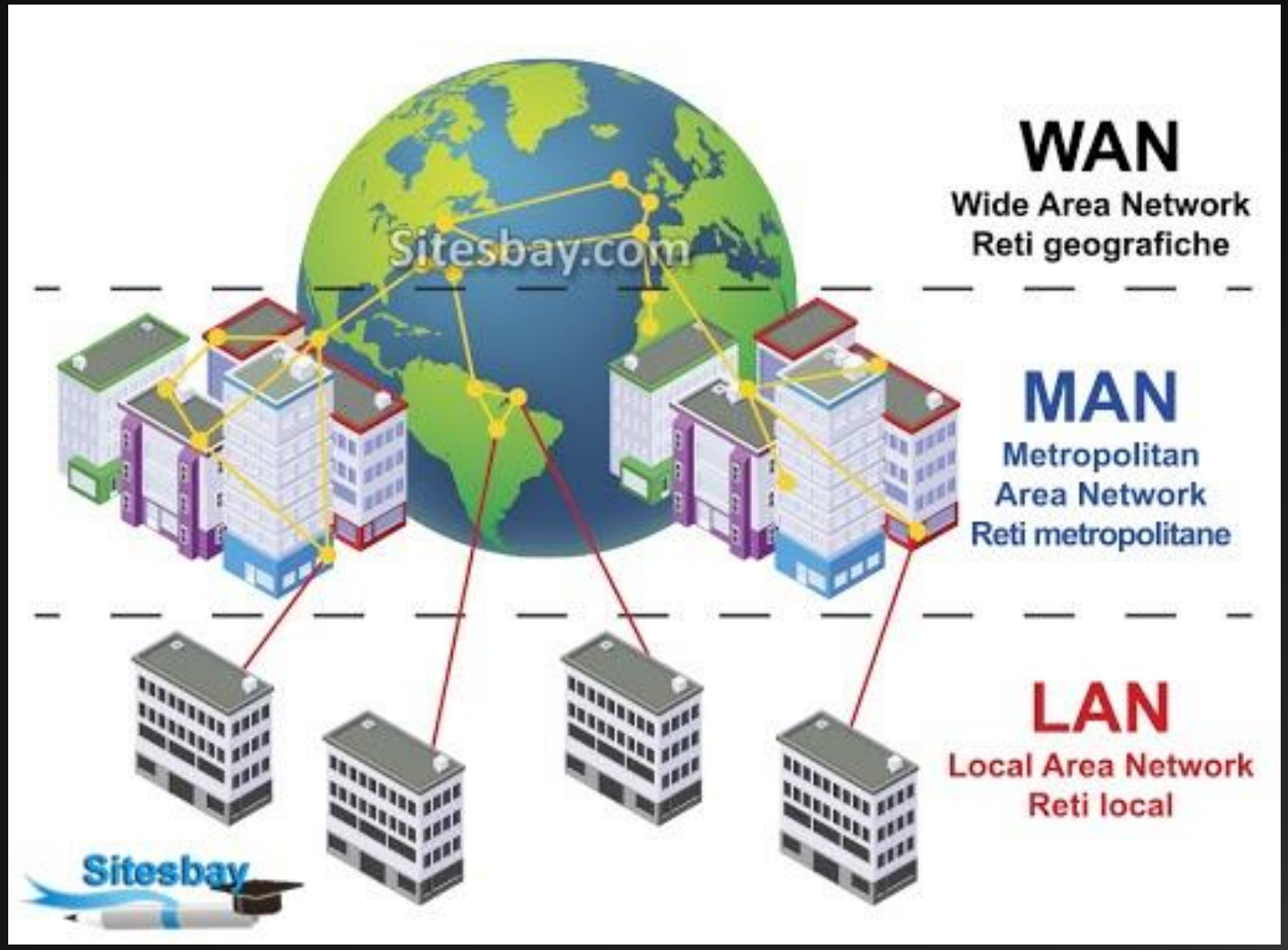


Data Structure

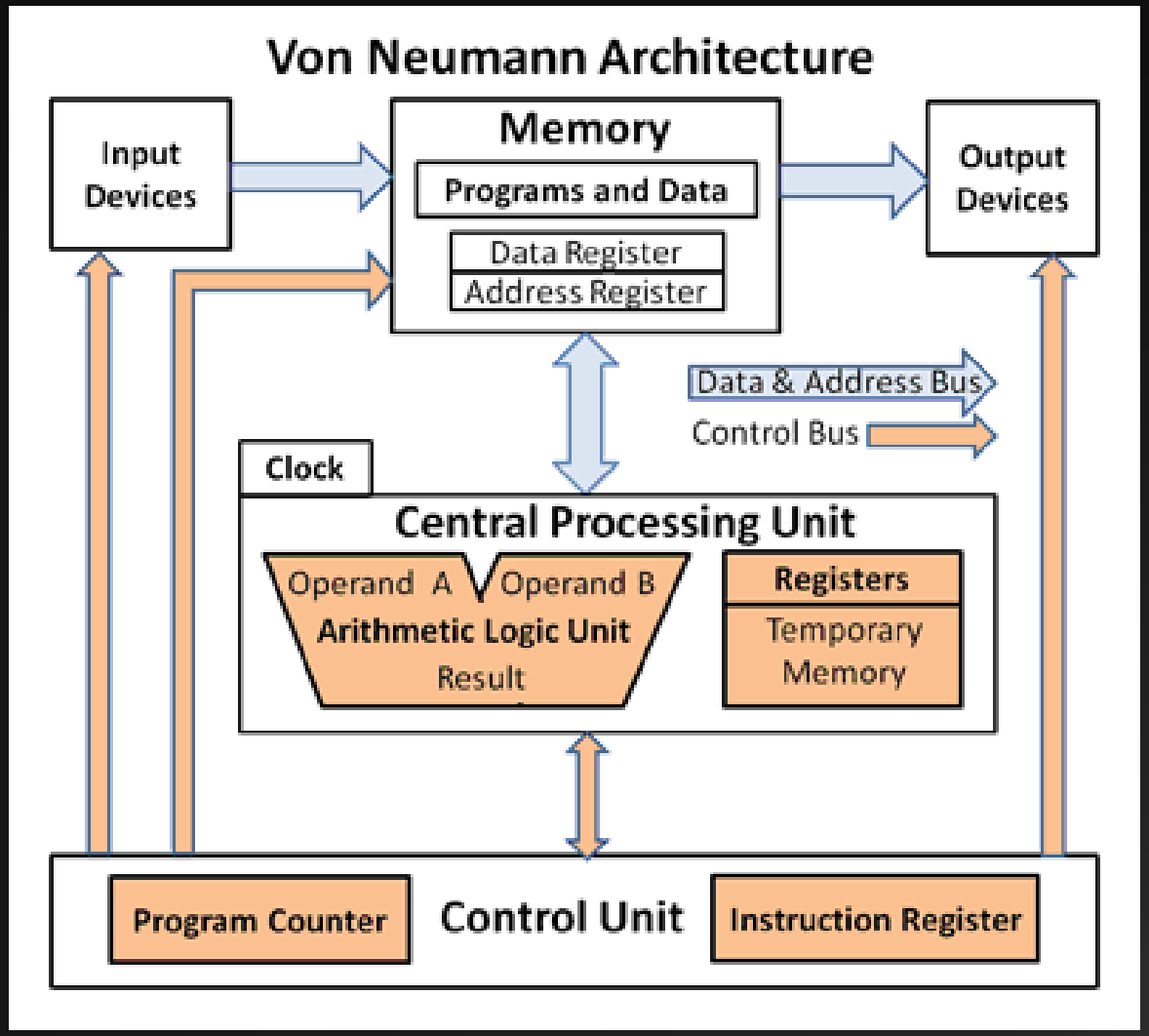

Types of Algorithms
Recursive Algorithm
Divide & Conquer Algorithm
Dynamic Programming Algorithm
Greedy Algorithm
Brute Force Algorithm
Backtracking Algorithm


Algorithm
GameTheory
Shortest
DFS
BFS
CP
Searching
Maximum
Search
Sorting
points
path
hull
ICPC
Matching
Exponentiation
Subsequence
Optimal
First
Longest
Flow
convex
vertex
intersection
Minimum
Hungarian
Bipartite
Modular
Sieve
graph
line
Distance
Vertices
Matrix
Method
cut
Union

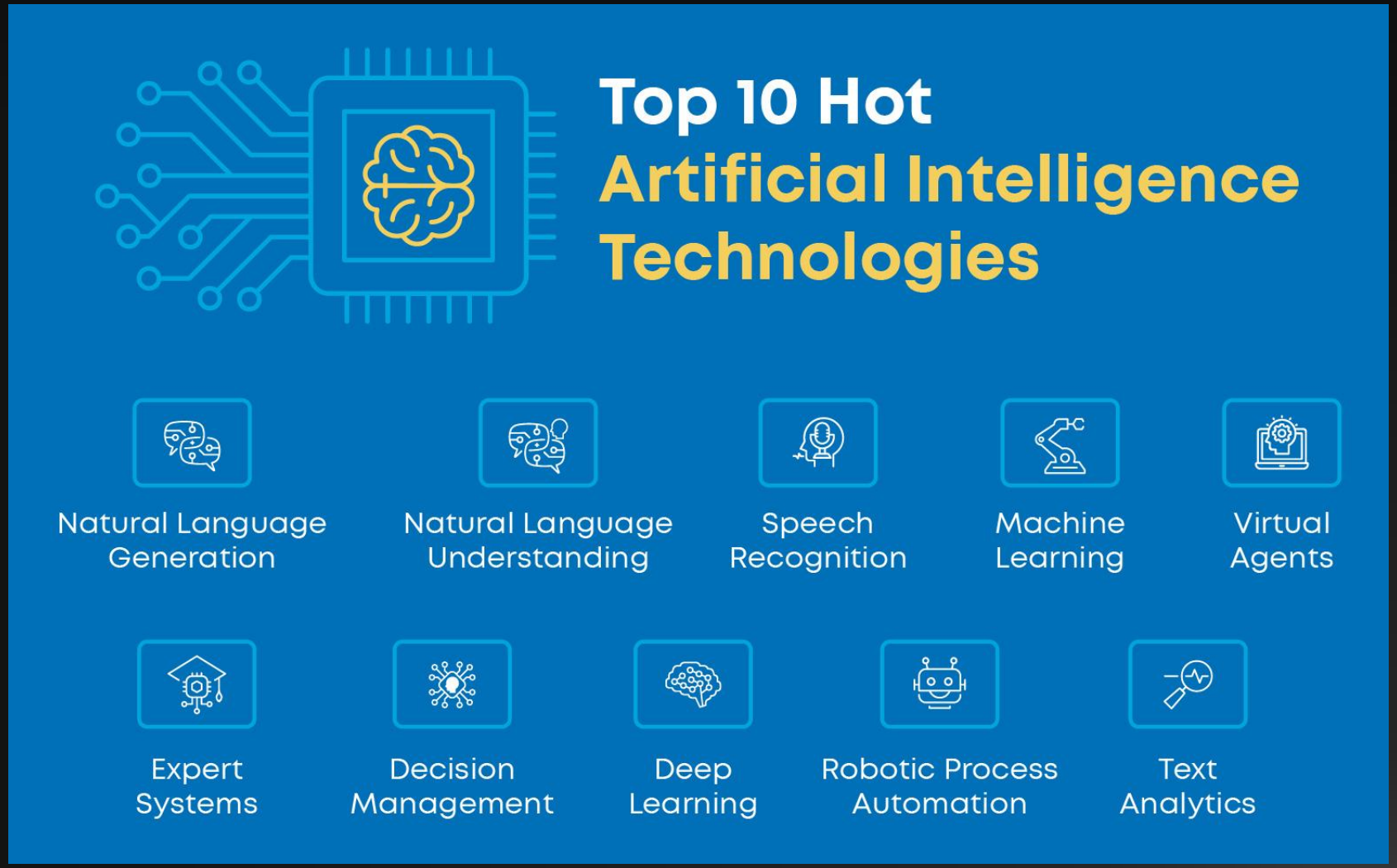
Algorithms



Computer Networks



Computer Architecture



Artificial Intelligence


年轻人，你渴望力量吗？

Dev C++

integrated development environment

Github

A programmer lovemost website

Github Desktop