Dynamic Programming!

Dynamic Programming is an optimization oven recursion. Whenever we see a recursive solution that has respected calls too same inputs we can optimize it using Dynamic Programming. The idea is to simply stone the results of subproblems, so that we do not have to recompute them when needed later. This simple optimization reduces time complexities from exponential to polynomial.

For example, if we write simple recursive solution for Pibonacci Numbers, we get exponential time complexity and it we optimize it by storing solutions to subproblems, time complexity reduces to linear

int fiblint n)

? if (n <= 1)

roztunntib(n-1)+dib(n-2);

Recursion: Exponential

Dynamic Programing: Linzan Dennie Breggermanning is the oftender simonga fon (1=2; i<=n; i++) ~ \$ \f[i] = f[i-i] + f[i-2]; en int about simoned friend in setalities nos Elderses 3 alt some floris of el sobi ecolos the stranger of Enji se emildocrados simple optimization radices lime complession exponential to polynomial. Ford transfer if the charter story solution for Mibonacci Humber & wi ti ban stimidmos mil loil asmogra Los of = noitatos primotes ded Li stirilgo asmil of combine Blindages smit-(Notal) Rid Ani PRESIDENT PROPERTY