Probabilistic Algorithms

Leaves Some of Its Decisions To Chance

Monte Carlo
Always Gives an answer
But not necessarily Correct
The probability of correctness goes up with time

Las Vegas
Never return an incorrect answer
But sometimes don't give any

Example of Numeric

Find Value of π

Draw a Square

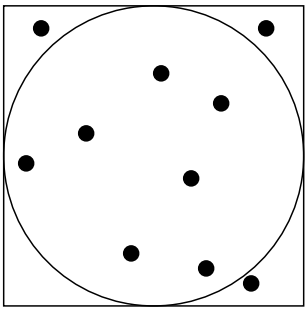
Inscribe a circle within it

Place it someone
and throw darts at it

Take the ratio
$4\frac{outside}{Total}$
outside = Number of darts landing outside the circle
Total = Number of darts thrown

This is an approximation of π

Computing Pi with a dart board



Circle = πr^2

Square = $4r^2$

State Space Search

8 Queens Problem

Place 8 Queens on a Chess Board
So that none can take the other

$\binom{64}{8} = 4426165368$ possibilities

Reduction:
Don't put queens on the same row or column
Check at the end if a possible solution
$8! = 40320$ possibilities
Backtracking: Solution after 2830 tries

Further Reduction:
If check as putting queens down
2057 possibilities
Backtracking: Solution after 114 tries

Las Vegas Approach

Observations:

There are many solutions
There seems to be no systematic way to place the queens

Algorithm:

Place each queen randomly in succession
making sure that no queen threatens the other

Failure results if a queen cannot be placed

Repeat until success

Las Vegas Approach

The number of nodes explored in case of success

$s = 9$

(counting the empty set)

The average number of nodes explored before failure

$e = 6.971...$

The probability of success

$p = 0.1293...$

The expected number of nodes to yield success

$s + \frac{(1-p)e}{p} = 55.9$

Better than the systematic approach

If combine with a backtracking
Not starting over on failure
can do even better