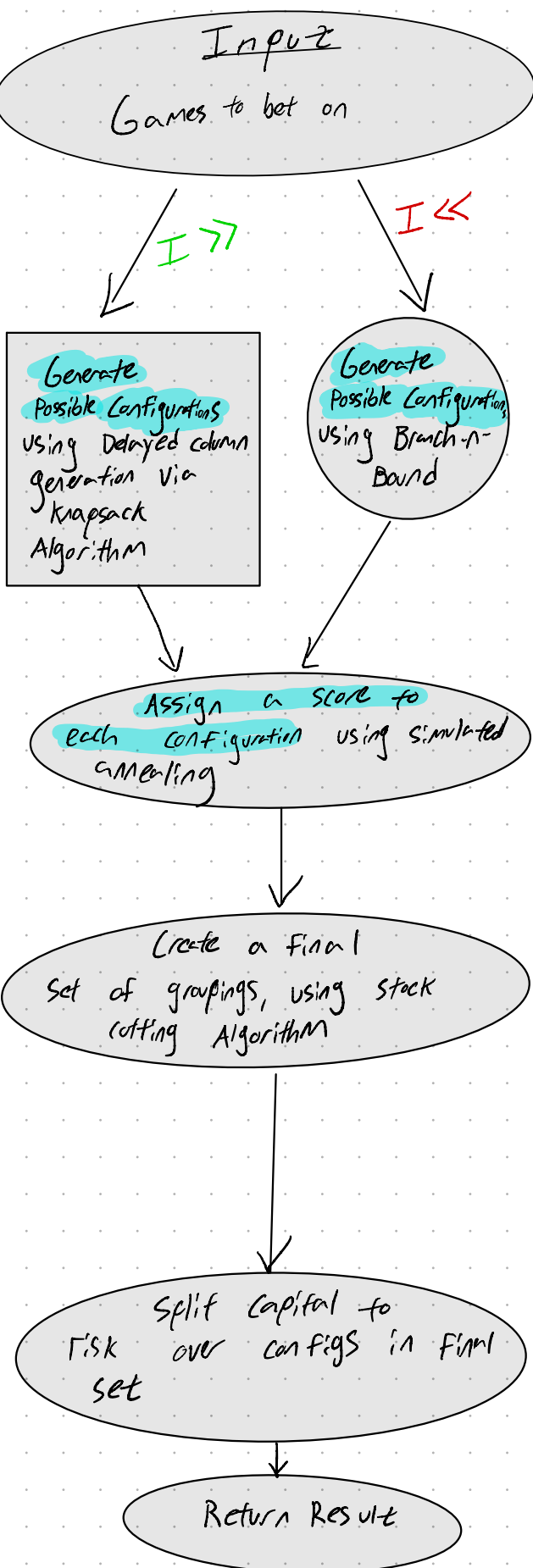


Stages



Details

- # of games
- Win % or odds for each game
- Total capital to risk

- Generate Possible Configurations of Bets using branch and bound if **I is small**, and the knapsack problem if **I is sufficiently large**

- Use Simulated Annealing to find best configuration
- Use each possible configuration to as a state for SA Algorithm
- Assign scores to each state based on how close they are to optimal

- Use stack cutting Algorithm to find an optimal set of configurations, to fulfill required number of bets

- Score of each configuration will be a 'width'

- Optimal set will have:
 - highest possible score
 - fulfill total # of bets
 - no duplicate configs

- Total capital will be split by:

$$C_c = \left(\frac{S_c}{S_T} \right) C_T$$

C_c = Config Capital

S_T = Score Total

S_c = Config score

C_T = Total Capital

Visualization

\$75

Dodgers vs.	Angels	D - 180
Cubs vs.	Cardinals	C - 100
Yankees vs.	Red Sox	Y + 100
A's vs.	Mariners	A + 250
Giants vs.	Phillies	P - 80

$X \neq X$

