Q-1

$x \in [a,b]$

m = number of decimals

i = number of iterations

I = max limit of iterations

$f(x)$ = func., $f'(x)$ = Derivative

→ If there are no guesses, start with $(a+b)/2$

Pseudo Code

$$\left.\begin{array}{l} X_{old} = 0 \\ i = 0 \end{array}\right\} O(1)$$

while (round(X_{old}, m) != round(X, m)) and ($i < I$):

$X_{old} = X$

$x = x - \underbrace{\frac{f(x)}{f'(x)}}_{F(m)}$

$O(\log m F(m))$

↳ $F(m)$ → Cost of calculating $f(x)/f'(x)$ in a given "m" number of decimals.

→ General process follows a recursive path with $O(\log n)$ and complexity depends on $f(x)$. So; $O(\log n F(m))$

Q-2

QuickSort's Complexity depends on the dispersion of numbers, basicly, degree of balance in positions of values. There could be 3 different complexity scales.

Worst Case

→ Most unbalanced position

→ Every value needs to be checked and swap prior elements.

$= n + (n-1) + (n-2) \ldots$

$= (n+1)(n/2) - 1$

$= O(n^2)$

Best Case

→ Partitions are in balanced position.

↳ Minimum effort.

→ $2 \cdot n/2 = n +$

$4 \cdot n/4 = n +$

$+ 8 \cdot n/8 = n$

$= O(n \log_2 n)$

Average Case

→ Partitions are in a relatively balanced position.

→ Complexity of average case is the same with best case.

$= O(n \log_2 n)$

Q-3

C_j = Credit volume

g_j = Risk grade

P_j = Expected Profit

G = max Risk capacity

C = max Credit Limit

$$\text{s.t.} \sum_{j=1}^{n} c_j/g_j\, x_j \le G \qquad \text{s.t.} \sum_{j=1}^{n} c_j x_j \le C$$

$$\max \sum_{j=1}^{n} P_j x_j$$

Value Function

$V[j, C, G] =$

$\max\left(V[j-1, C, G],\ P_j + V[j-1, C - C_j, G - \frac{c_j}{g_j}]\right)$

Complexity

$O(nGC)$

Q-4

P_{ijk} → i banktan j banka k tipi trans

P_{ij} → i banktan j banka trans tipi sayısı

$$\text{s.t.} \sum_{j=1}^{n} \left[\sum_{k=1}^{P_{ij}} P_{ijk} X_{ijk} - \sum_{k=1}^{P_{ij}} P_{jik} X_{jik}\right] \le C_i \quad \forall i = 1, 2 \ldots n$$

$$\max \sum_i \sum_j \sum_{k=1}^{P_{ij}} P_{ijk} X_{ijk}$$

Value function

$V[i, j, k, P_{ijk} - P_{jik}] =$

$\max(V[i, j, k-1, C_i], P_{ijk} + V[i, j, k-1, (P_{ijk} - P_{jik})])$

Complexity

$O(nC)$

Scanned with CamScanner