

Maratona de Programação Semana de Informática 2022

André G. Santos¹ Salles V. G. Magalhães¹

¹Departamento de Informática
Universidade Federal de Viçosa (UFV), Brazil

Semana de Informática, 2022

G. Campina Grande (14/15)¹

Resumo

- Ler o nome da cidade (2 palavras) onde será realizada a maratona
- Ler uma lista de nomes de outras cidades e verificar quais causam confusão com a cidade correta (1ª ou 2ª palavra igual à correta)

Solução

- Simplesmente fazer o que se pede

¹quantos times passaram a questão e número de submissões

Resumo

- Ler um valor e verificar se pode ser pago com transações PIX = 3.14
- As transações podem ser de pagamento ou de troco

Solução

- Note que se for possível com troco, é possível sem troco
- O valor é dado em centavos, então basta verificar se é múltiplo de 314

D. Eleição para presidente (12/20)

Resumo

- Eleição com dois candidatos, Capinaro 22 e Capissilva 13
- As capivaras eleitoras memorizam a soma dos dígitos do candidato
- Ler os votos e informar quem venceu, ou ninguém ou indeterminado

Solução

- Note que, para ambos candidatos, a soma dos dígitos é 4
- Portanto, pela memória das capivaras, não há como diferenciá-los
- Basta contar o número de votos 4 (restante é inválido)
- Se a maioria não é 4, ninguém venceu; senão, indeterminado

C. Eleição para reitor (11/32)

Resumo

- Eleição com cinco partidos: -1, 0, 100111001, 2.71828..., 3.1415...
- Os votos nos partidos $e = 2.71828\dots$ e $\pi = 3.1415\dots$ podem ser informados por 1 a 15 dígitos
- Informar qual venceu (ou se houve empate)

Solução

- Note que basta verificar o primeiro caractere: - 0 1 2 3
- Criar um array de contadores ou um dicionário e contar os votos
- Encontrar o maior valor e verificar ele aparece mais de uma vez

B. Data da maratona (10/22)

Resumo

- Ler as datas de N provas de cálculo e determinar a data da maratona
- A data da maratona é o dia com maior número de provas de cálculo
- Em caso de empate, a data mais próxima do início do ano

Solução

- Note que o número K (de Cálculo K) é irrelevante
- Ler as datas e contar as ocorrências de cada uma
- Mais fácil com dicionário (`map` em C++, `dict` em Python)
 - a data pode ser lida como string e indexada como tal
- Alternativamente pode-se usar uma matriz 12×31 de contadores
- Ou um vetor de contadores indexado, p.ex., por `mês × 100 + dia`

I. Meia pizza (9/28)

Resumo

- Ler os preços de P sabores de pizza e os pedidos de N alunos
- Os pedidos são feitos dois a dois (meia-pizza para cada aluno)
- O preço de uma pizza de dois sabores é o do sabor mais caro
- Determinar o valor mínimo para comprar todos os sabores pedidos

Solução $O(N \log N)$ – método guloso

- Juntar o sabor mais caro com o 2º mais caro, e assim por diante:
 - Criar um vetor com os valores dos N sabores pedidos
 - Se N for ímpar, acrescentar mais um sabor, o mais barato
 - Ordenar de forma decrescente
 - Somar os valores de 2 em 2
- Assim, pagará apenas o mínimo necessário

J. Capibonacci e Pizzibonacci (2/31)

Resumo

- Determinar o último dígito do N -ésimo número de Fibonacci
- Dificuldade: N pode valer até 10^{1000}

Solução $O(N)$

- Calcular $F_i = F_{i-1} + F_{i-2}$ para $i = 0 \dots N$: **Time limit**

Solução $O(\log_{10} N)$

- Note que a sequência de último dígito deve ser periódica, i.e., deve se repetir, pois há apenas 10×10 combinações possíveis para F_{i-1} e F_{i-2}
- Um pré-processamento revelaria que $F_{60} = 0$ e $F_{61} = 1$
- Logo, $F_{60} = F_0$, $F_{61} = F_1$, $F_{62} = F_{61} + F_{60} = F_1 + F_0 = F_2$, ..., $F_n = F_{n\%60}$
- Basta calcular os 60 primeiros Fibonacci e escrever o $(n\%60)$ -ésimo valor
- E como calcular o resto de N por 60 sendo N um número gigantesco?
 - de forma iterativa à medida que lê os dígitos, semelhante ao processo manual de divisão (obs.: em Python basta ler N e calcular $N\%60$)

Resumo

- Grafo em que cada aresta tem tempo (t_i) e número de solavancos (s_i)
- O Velho do Lago quer o melhor caminho até o hospital:
o caminho de até T minutos com menos solavancos no pior trecho

Solução 1: busca binária + caminho mínimo simples

- Seja S o máximo de solavancos suportado ($0 \leq S \leq 1000000000$)
- Encontrar o caminho de tempo mínimo no grafo com arestas $s_i \leq S$
- Encontrar o valor ideal de S por busca binária
 - Se tempo mínimo no grafo $\leq T$ diminuir S ; senão, aumentar S
- São no máximo $\log_2 1000000000 = 30$ resoluções de caminho mínimo

Solução 2: caminho mínimo com tuplas

- Manter (*tempo*, *max solavancos*) no caminho, não apenas o tempo

K. Baking Bread (0/5)

Resumo

- Contar de quantas formas N pães podem ser dispostos em fileiras
- Cada fileira deve ter menos pães que a fileira abaixo

Solução $O(N^2)$ – programação dinâmica

- Seja $P[i, j]$ o número de maneiras de i pães com j na base
- Queremos o valor de $P[n, 1] + P[n, 2] + \dots + P[n, n]$
- Note que, para i pães, com j na base, sobram $i - j$ para cima
- Logo, $P[i, j]$ é o número de maneiras de $i - j$ pães com $k < j$ na base

$$P[i, j] = \sum_{k < j} P[i - j, k]$$

com $P[i, j] = 0$ se $i < j$.

F. Rodizio de mini pedaços de mini pizza (0/4)

Resumo

- N mini pizzas: servidas no tempo t_i , de qualidade q_i , consumida em c_i min
- Uma mini pizza por vez: se comer i , não pode aceitar j se $t_j > t_i - c_i$
- Escolher até K mini pizzas para maximizar qualidade total

Solução $O(N \log N + NK)$ – programação dinâmica

- Ordenar as mini pizzas pelo tempo que terminaria de comer $f_i = t_i - c_i$
- Seja $p(i)$ a última que termina a tempo de comer i : $p(i) = \max_j f_j \geq t_i$
- A qualidade máxima $Q[i]$ até a mini pizza i é o máximo entre:
 - comer i : $q_i + Q[p(i)] \rightarrow$ qualidade dela + max anteriores que pode comer
 - não comer i : $Q[i - 1] \rightarrow$ qualidade max das anteriores
- Para limitar K : seja $Q[i][k]$ qualidade máxima até i com k mini pizzas
 - $Q[i][k] = \max\{q_i + Q[p(i)][k - 1], Q[i - 1][k]\}$

E. PCV - O Problema da Capivara Visitante (0/0)

Resumo

- N lagoas interligadas por riachos, cada um com um custo para passar
- Achar o custo do tour de custo mínimo que visita uma vez cada lagoa
- \Rightarrow Problema do caixeiro viajante
- Obs.:
 - Cada lagoa se conecta a no máximo $N/3$ lagoas por riachos com jacarés
 - As conexões com as demais lagoas são por riachos sem jacarés, custo 1

Solução

- Note que o grafo de arestas de custo 1 satisfaz o Teorema de Dirac:
Um grafo com $n \geq 3$ vértices é hamiltoniano se cada vértice tem grau $\geq n/2$
- Então, há solução sem jacarés...