Tópicos Especiais em Algoritmos
Divisão e Conquista
Editorial

Daniel Saad Nogueira Nunes

VI Maratona de Programação do IFB: AC ou WA?

Sejam F_k e S_k, respectivamente, as k-ésimas strings de Fibonacci esperada e a gerada por Tarcísio. O objetivo é verificar se o i-ésimo caractere de F_k e S_k são iguais, em que F_k e S_k são strings suficientemente grandes para englobar o índice i, conforme o enunciado do problema. Sem gerar as strings, é possível determinar qual é este caractere ao examinar as definições de F_k e S_k. Temos que:

$$F_k = \begin{cases} B, & k = 1 \\ A, & k = 2 \\ F_{k-1} \cdot F_{k-2}, & k > 2 \end{cases}$$

$$S_k = \begin{cases} B, & k = 1 \\ A, & k = 2 \\ S_{k-2} \cdot S_{k-1}, & k > 2 \end{cases}$$

Se soubermos $|F_{k-1}|$, $|F_{k-2}|$, $|S_{k-1}|$ e $|S_{k-2}|$, isto é, o tamanho das substrings que geram F_k e S_k, podemos reduzir o problema até chegar no caso base e encontrar o caractere desejado, já que cada string é composta da concatenação de duas anteriores.

Para F_k, o i-ésimo caractere pode ser obtido através do Algoritmo 1.

Algorithm 1: $\textsc{f}(n, i)$

if $n = 1$ **then**
    return A
if $n = 2$ **then**
    return B
if $i \leq |F_{n-1}|$ **then**
    return $f(n-1, i)$
return $\textsc{f}(n-2, i - |\textsc{f}_{n-1}|)$

O algoritmo para encontrar o i-ésimo caractere de S_k é similar, mas adequando à definição de S_k.

Para descobrir o tamanho $|F_n|$ ou $|S_n|$, basta pré-computar a sequência de Fibonacci até que se atinja um valor superior à 10^{18}. Para descobrir o valor de k em F_k ou S_k, basta realizar uma busca binária sobre a sequência de Fibonacci pré-computada.

Complexidade

Para cada pergunta sobre um índice i, a complexidade é $O(\log_\Phi(i))$, em que Φ é a razão áurea.

Detalhes de Implementação

Inteiros de 64-bits devem ser utilizados para comportar a magnitude dos valores.

Seletiva UnB 2019: Presente de Dia das Mães

Este problema pode ser resolvido utilizando a técnica de busca binária na resposta sobre o intervalo $[0, v_{max}]$, em que v_{max} corresponde ao maior número de chocolates considerando cada caixa.

Seja x a tentativa escolhida pela busca binária. Cada caixa $V[i]$ consegue atender exatamente $\lfloor \frac{V[i]}{x} \rfloor$ mães. Ao realizar o procedimento para cada caixa, basta verificar se o número de mães M foi atendido ou não e ajustar o intervalo da busca binária para cada iteração.

Detalhes de Implementação

Deve-se atentar para a divisão por zero, já que pode haver caixas vazias.

Complexidade

A complexidade da solução é $O(n \lg(v_{max}))$.

Codeforces 1612C: Chat Ban

Mais uma vez, utilizamos busca binária na resposta sobre o intervalo $[1, 2k-1]$. Para cada tentativa com valor x, devemos verificar quantos emoticons possuímos até a x-ésima linha. Seja $S(y)$ a soma da P.A. de razão 1 no intervalo inteiro $[1, y]$. Temos dois casos:

- $x \leq k$: o número de emoticons é exatamente $S(k)$.
- $x > k$: podemos decompor a figura em duas partes:
  - $[1, k]$: o número de emoticons é igual a (k).
  - $[k+1, 2k-1]$: o número de emoticons é igual a $S(k-1)$, i.e., as $k-1$ últimas linhas.

  Contudo, precisamos descontar algumas linhas do final da figura quando $x < 2k-1$, e o desconto é exatamente a soma da P.A de razão 1 sobre o intervalo $[1, 2k-1-x] = S(2k-1-x)$. Desta forma, o número de emoticons no segundo caso é dado por: $S(k)+S(k-1)-S(2k-1-x)$.

Após calcular o número de emoticons, é possível ajustar os intervalos da busca binária e salvar a resposta de acordo com o número de emoticons produzidos ao enviar x mensagens.

Complexidade

A complexidade da solução é $O(\lg(2k-1))$, haja vista que $S(y)$ pode ser calculado em tempo constante, para qualquer y, através da fórmula $S(y) = y(y+1)/2$.

Detalhes de Implementação

Inteiros de 64-bits devem ser utilizados na solução para poder comportar os possíveis valores de entrada.

Maratona de Programação contra o COVID-19: Encontrando o Pico

Dada a função logística $f(x) = c/(1 + ae^{-bx})$, o que o problema quer é encontrar o x tal que $f'(x)$ é o maior possível. Isto pode ser feito de pelo menos duas formas:

1. Analítica.
2. Numérica.

Como se trata de uma lista sobre divisão e conquista, recomenda-se que a questão seja realizada através do método numérico para fixação de conceitos, mas as duas soluções serão apresentadas a seguir.

Solução Analítica

Queremos o ponto em x em que $f'(x)$ alcança o maior valor. Isto pode ser obtido simplesmente encontrando a raiz para $f''(x)$, visto que $f'(x)$ é uma função unimodal. Basta derivar $f(x)$ duas vezes para encontrar $f''(x)$ e a fórmula fechada para encontrar sua raiz dará o valor esperando de x que maximiza $f'(x)$. Uma vez encontrado o ponto x, basta imprimir o par $(x, f'(x))$. Detalha-se a resolução abaixo.

A derivada da função logística é:

$$f'(x) = \frac{abce^{bx}}{(e^{bx} + a)^2}$$

O objetivo é achar o valor máximo de $f'(x)$. O ponto de valor máximo é justamente o ponto em que a segunda derivada da função logística é 0, em outras palavras, estamos buscando x tal que $f''(x) = 0$.

Calculando a segunda derivada, temos:

$$f''(x) = -\frac{ab^2ce^{bx}(e^{bx} - a)}{(e^{bx} + a)^3}$$

Igualando $f''(x)$ a zero, obtemos que a raiz é justamente $x = \ln(a)/b$, este é o ponto em que $f'(x)$ é o maior possível.

Ao calcular $f'(\frac{\ln(a)}{b})$, obteremos $\frac{bc}{4}$, que é o número máximo de novos infectados.

Solução Numérica

A função derivada é uma função unimodal, que cresce, chega a um pico, e logo diminui. O gráfico da Figura 1 corresponde à derivada do primeiro exemplo de teste.

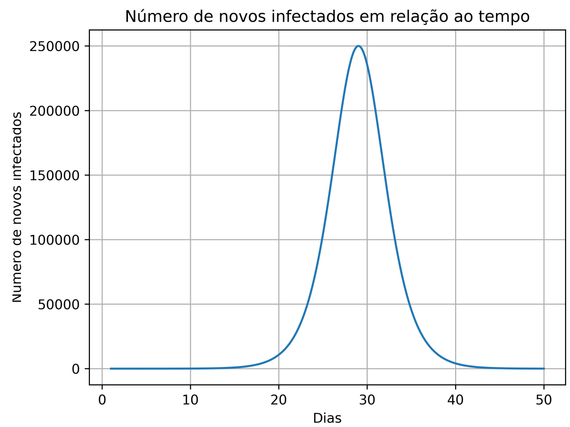

Figura 1: $f'(x)$ para o primeiro caso de teste.

Desta forma, para achar o momento de pico e este número propriamente dito, basta aplicar a técnica de **busca ternária** sobre a derivada [Wik].

Para calcular o valor desta derivada em um ponto x específico durante a busca ternária, podemos usar a definição de derivada através de limite:

$$\lim_{\epsilon \to 0} \frac{f(x+\epsilon) - f(x)}{\epsilon}$$

Com ϵ na prática sendo um valor minúsculo, como por exemplo 10^{-5}.

É importante também escolher um intervalo de busca grande o suficiente na busca ternária para garantir que a solução da busca ternária seja ótima.

Referências

[Wik] Wikipedia, *Ternary search*, `https://en.wikipedia.org/wiki/Ternary_search`, acessado em dezembro/2021.