SCC-0218 PROJETO 3
FIBONACCI EM $O(\log_2 n)$

1. Subsídios

Nesta seção serão apresentados os subsídios necessários para confecção do trabalho.

1.1. Sequências de combinação linear. Suponha que temos uma sequência S, sendo que os seus k primeiros elementos $S_0, S_1, \ldots, S_{k-1}$ são casos base, e qualquer outro é uma combinação linear dos k anteriores: $S_{i+1} = \alpha_1 S_i + \alpha_2 S_{i-1} + \cdots + \alpha_k S_{i-k}$. Podemos calcular S_{i+1} matricialmente, conforme a Equação 1.

$$\begin{bmatrix} S_{i+1} \\ S_i \\ S_{i-1} \\ \vdots \\ S_{i-k+1} \end{bmatrix} = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \ldots & \alpha_k \\ 1 & 0 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & 0 & \ldots & 0 \end{bmatrix} \begin{bmatrix} S_i \\ S_{i-1} \\ S_{i-2} \\ \vdots \\ S_{i-k} \end{bmatrix} \tag{1}$$

Para exemplificar, vamos supor a sequência $S = 1, 2, 3, 10, 22, \ldots$, na qual o $S_0 = 1$, $S_1 = 2$, $S_2 = 3$ e $S_{i+1} = 1S_i + 2S_{i-1} + 3S_{i-2}$. Podemos calcular $S_3 = 10$ conforme a Equação 2.

$$\begin{bmatrix} 10 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} S_3 \\ S_2 \\ S_1 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} S_2 \\ S_1 \\ S_0 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} = A \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} \tag{2}$$

Podemos calcular $S_4 = 22$ conforme a Equação 3.

$$\begin{bmatrix} 22 \\ 10 \\ 3 \end{bmatrix} = \begin{bmatrix} S_4 \\ S_3 \\ S_2 \end{bmatrix} = A \times \begin{bmatrix} S_3 \\ S_2 \\ S_1 \end{bmatrix} = A \times A \times \begin{bmatrix} S_2 \\ S_1 \\ S_0 \end{bmatrix} = A^2 \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} \tag{3}$$

Generalizando, obtemos a Equação 4.

$$\begin{bmatrix} S_{i+1} \\ S_i \\ S_{i-1} \end{bmatrix} = A^{i+1} \times \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} \tag{4}$$

1.2. **Potenciação em $O(\log_2 n)$.** Suponha que desejamos calcular x^n, sendo n um número natural positivo. Podemos utilizar a divisão e conquista para reduzir o número de multiplicações a $O(\log_2 n)$. Para isso, basta utilizar a fórmula recursiva da Equação 5. A demonstração da complexidade fica à cargo do aluno, e é esperada no relatório deste trabalho.

$$x^1 = x \tag{5a}$$

$$x^2 = x \times x \tag{5b}$$

$$x^n = \left(x^{n/2}\right)^2 \text{ se } n \text{ é par} \tag{5c}$$

$$x^n = x\left(x^{\lfloor n/2 \rfloor}\right)^2 \text{ se } n \text{ é ímpar} \tag{5d}$$

2. Descrição do trabalho

Considere a sequência $F = 1, 1, 2, 3, 5, \ldots$ na qual $F_0 = 1$, $F_1 = 1$ e $F_n = F_{n-1} + F_{n-2}$. Esta sequência é conhecida como *Sequência de Fibonacci*, e se mostra presente em inúmeros fenômenos da natureza, sendo de grande interesse de estudo.

Para este trabalho, utilizando as ideias apresentadas anteriormente, o aluno deve desenvolver um programa que receba pela entrada padrão um inteiro n tal que $0 \leq n \leq 10^{11}$, e imprima na saída padrão o valor $F_n \mod 10^6$.

Um curto relatório justificando a complexidade do algoritmo e relacionando-a à técnica de divisão e conquista deve ser entregue, junto ao código, até o dia **25 de Novembro**, no escaninho do Tidia.

A correção do código valerá 50% da nota do trabalho, enquanto o texto apresentado no relatório valerá 50%.

Dica: $ab \mod M = (a \mod M)(b \mod M) \mod M$