Atividade de Matemática Discreta 2 – Respostas da Questão 04

Aluno: Rodrigo Henrique Donato de Souza

Matrícula: 241012374

Aluno: Raphaela Guimarães de Araujo dos Santos

Matrícula: 190116072

1.

Código Preenchido:

- **Linha 10:** while (b != 0) {
- **Linha 14:** b = resto;
- **Linha 23:** if (mdcComPassos(a, m) != 1)
- **Linha 36:** x1 += m0;
- **Linha 45:** if (exp % 2 == 1)
- **Linha 72:** int inverso = inversoModular(G, Zn);
- **Linha 78:** int resultado = powMod(a, x, n1);

Saída do Programa com os Valores Fornecidos
c:\MD2\codigos\output>.\"questao4.exe"

Insira H: 7

Insira G: 3

Insira Zn: 11

Insira x: 10

Insira n1: 13

Algoritmo de Euclides: 3 mod 11 = 3

Algoritmo de Euclides: 11 mod 3 = 2

Algoritmo de Euclides: 3 mod 2 = 1

Algoritmo de Euclides: 2 mod 1 = 0

Substituindo, temos que o inverso de 3 em mod 11 é 4.

Fazendo a multiplicação modular: (7 * 4) mod 11 = 6

Sendo 4 o inverso de 3

Valor final da congruência: 4

c:\MD2\codigos\output>

2.

Análise das Afirmativas

- **(V)** O algoritmo de Euclides estendido é utilizado para calcular o inverso modular de um número.
- **(F)** Se mdc(G, Zn) ≠ 1, o programa ainda consegue encontrar o inverso de G em Zn.
- **(V)** A operação (H * inverso) % Zn representa a divisão modular de H por G.
- **(F)** Se n1 for primo, o código aplica o Pequeno Teorema de Fermat para simplificar o cálculo de a^x mod n1.
- **(F)** A função powMod implementa o cálculo de potência modular utilizando multiplicações diretas sem otimização.
- **(V)** Quando o resultado do inverso é negativo, o código ajusta o valor somando o módulo m0.
- **(F)** O cálculo de fi(n1) (função totiente de Euler) é utilizado apenas quando n1 não é primo.