# Algoritmo Trabalho Ahmes

## Introdução a Arquitetura de Computadores

Nome: João Pedro Silveira e Silva

Matrícula: 00303397

1. Zera os valores das variáveis que armazenaram o resultado final (posição 130 e 131 da memória em decimal)
2. Salva o valor de “n” em uma variável temporária “vd” e “vda” que serão utilizadas como possíveis divisores (“vd” será utilizado como temporário para realizar a divisão por subtrações sucessivas enquanto “vda” armazenará o valor do divisor atual intacto).
3. Verifica se o possível divisor “vd” é realmente um divisor:
   1. Subtrai o valor de “vd” em “n”
   2. Se o resultado for Zero é divisor, então obtém o valor do divisor elevado a “x”:
      1. Se o expoente “x” for ZERO soma um ao resultado final.
      2. Se não, subtrai 1 do valor do expoente “e” e só resultado for ZERO soma o valor do divisor atual “vda” no resultado final.
      3. Caso não entre nos casos “3.2.1” e “3.2.2” considera-se que o expoente é 2. Então multiplica a base por ela mesma. Inicializa um contador com o valor 8.
         1. Rotaciona o valor da base para a direita.
         2. Soma o valor da base no resultado mais significativo.
         3. Rotaciona o resultado mais significativo para a direita.
         4. Rotaciona o resultado menos significativo para a direita.
         5. Subtrai um do contador, se ele for ZERO soma o resultado atual da potenciação com o resultado final. Se não prossegue novamente com a etapa “3.2.3.1”.
   3. Se o resultado for Negativo não é divisor, então lê o valor de “vda” subtrai um e caso não dê ZERO retorna para o passo “3.1” com a nova tentativa de divisor. Em caso de resultado ZERO significa que o programa já realizou o teste com todos os divisores possíveis então finaliza.
   4. Caso não entre nos casos “3.2” e “3.3”, ou seja, o resto da subtração dê um valor positivo, então soma o valor de “vda” em “vd” (obtendo um novo múltiplo do divisor atual) e retorna para o passo “3.1” com este novo valor para “vd”.