Proceeding Series of the Brazilian Society of Applied and Computational Mathematics, Vol. 3, N. 1, 2015.
Trabalho apresentado no XXXV CNMAC, Natal-RN, 2014.

O Problema dos 3 Potes d'água

César Guilherme de Almeida,
Faculdade de Matemática, UFU,
38408-100, Uberlândia, MG
E-mail: cesargui@ufu.br,

Resumo: *Neste trabalho será apresentado o problema dos três potes d'água e sua generalização. O problema é o seguinte: dados 3 potes com capacidades volumétricas* $c_3 = 8, c_2 = 5$ *e* $c_1 = 3$, *como dividir, com um número mínimo de movimentos, a água do pote maior, que está cheio, entre os dois potes menores, que estão vazios, de forma que as quantidades* p_3, p_2 *e* p_1 *dos potes maior (pote 3), intermediário (pote 2) e menor (pote 1), respectivamente, satisfaçam: (i)* $p_3 = p_2 + p_1$ *e (ii)* $p_3 = \frac{c_3}{2}$. *Considerando-se* c_3 *um número par qualquer e* c_3, c_2 *e* c_1 *dois a dois primos entre si, o problema pode ser generalizado e a sua solução é obtida com a quantidade mínima de movimentos igual a* $c_3 - 1$*; partindo-se da configuração inicial:* $p_3 = c_3, p_2 = 0$ *e* $p_1 = 0$.

Palavras-chave: *algoritmo de Euclides, teoria dos números, códigos computacionais.*

1 Introdução

A primeira vez que tive contato com o problema dos 3 potes foi na minha adolescência. Um tio que gostava de matemática apresentou o problema para os sobrinhos. "Um sujeito voltando de uma bica d'água carregava todo contente o seu pote completamente cheio com os últimos 8 litros do líquido precioso, que fora liberado naqueles tempos de estiagem severa. No caminho, encontrou dois amigos que carregavam tristonhos os seus potes vazios. Um dos potes tinha capacidade para 3 litros e o outro tinha capacidade para 5 litros. Honesto, amigável e justo, o sujeito resolveu dividir a água entre os companheiros. Decidiram, em comum acordo, que a divisão seria a seguinte: 4 litros deveriam ficar no pote com capacidade para 8 litros; 1 litro deveria ficar no pote com capacidade para 5 litros e 3 litros deveriam ficar no pote com capacidade para 3 litros. Para que a divisão fosse justa e agradasse a todos, os amigos que estavam com os potes menores resolveram permutar os mesmos. Explique como esta divisão foi feita, sabendo-se que os potes, que não continham nenhuma marcação volumétrica, eram os únicos referenciais que eles possuiam para fazer as medições."

A solução do problema anterior, que tem a menor quantidade de movimentos (despejar a água de um pote para o outro), pode ser representada pelas seguintes configurações:

- $p_3 = 8, \quad p_2 = 0, \quad p_1 = 0;$ (0)
- $p_3 = 5, \quad p_2 = 0, \quad p_1 = 3;$ (1)
- $p_3 = 5, \quad p_2 = 3, \quad p_1 = 0;$ (2)
- $p_3 = 2, \quad p_2 = 3, \quad p_1 = 3;$ (3)
- $p_3 = 2, \quad p_2 = 5, \quad p_1 = 1;$ (4)
- $p_3 = 7, \quad p_2 = 0, \quad p_1 = 1;$ (5)

- $p_3 = 7, \qquad p_2 = 1, \qquad p_1 = 0;$ (6)
- $p_3 = 4, \qquad p_2 = 1, \qquad p_1 = 3.$ (7)

Observe que as quantidades p_3, p_2 e p_1 dos potes maior, intermediário e menor, respectivamente, satisfazem: $p_3 + p_2 + p_1 = c_3$, onde c_3 é a capacidade volumétrica do pote maior. Se as capacidades volumétricas dos potes menores forem expressas por c_2 e c_1, com $c_1 < c_2$ e $c_3 = c_2 + c_1$, então, na última configuração, será obtido $p_3 = p_2 + p_1$, onde $p_3 = \frac{c_3}{2}$.

Uma outra solução do problema, com a mesma quantidade de movimentos, pode ser obtida utilizando-se os dados anteriores. Basta executar, em cada linha anterior (de 1 a 7), os seguinte cálculos: $c_3 - p_3$, $c_2 - p_2$ e $c_1 - p_1$. Obtendo-se as novas configurações:

- $p_3 = 8, \qquad p_2 = 0, \qquad p_1 = 0$ (0)
- $p_3 = 3, \qquad p_2 = 5, \qquad p_1 = 0$ (1)
- $p_3 = 3, \qquad p_2 = 2, \qquad p_1 = 3$ (2)
- $p_3 = 6, \qquad p_2 = 2, \qquad p_1 = 0$ (3)
- $p_3 = 6, \qquad p_2 = 0, \qquad p_1 = 2$ (4)
- $p_3 = 1, \qquad p_2 = 5, \qquad p_1 = 2$ (5)
- $p_3 = 1, \qquad p_2 = 4, \qquad p_1 = 3$ (6)
- $p_3 = 4, \qquad p_2 = 4, \qquad p_1 = 0$ (7)

2 Definições e resultados preliminares

Definição 2.1 *Uma configuração inicial de passo zero será representada por* IC_0: $p_3 = 0,\ p_2 = c_2,\ p_1 = c_1$ *ou* IC_0: $p_3 = c3,\ p_2 = 0,\ p_1 = 0$. *Uma configuração inicial de passo 1 será representada por* IC_1: $p_3 = c_2,\ p_2 = 0,\ p_1 = c_1$ *ou* IC_1: $p_3 = c_1,\ p_2 = c_2,\ p_1 = 0$.

Definição 2.2 *Os movimentos permitidos entre dois potes são aqueles que conduzem a uma das seguintes situações: ou um dos potes fica vazio ou um dos potes é preenchido completamente.*

Observação: As configurações C: $p_3,\ p_2,\ p_1$ e $\tilde{C}$: $\tilde{p_3},\ \tilde{p_2},\ \tilde{p_1}$, onde $\tilde{C}$ foi obtida de C através do desague do pote 3 no pote 2, não podem ocorrer, se forem considerados $c_2 > \tilde{p_2} > p_2 \neq 0$; $p_3 > \tilde{p_3} > 0$ e $p_1 = \tilde{p_1}$. Note que: $p_3 = \tilde{p_3} + (p_3 - \tilde{p_3})$ e $\tilde{p_2} = p_2 + (p_3 - \tilde{p_3})$. Assim, para fazer o desague do pote 3 no pote 2, a quantidade $p_3 - \tilde{p_3} > 0$ teria que ser separada da quantidade $\tilde{p_3} > 0$. Além disto, seria impossível reverter o movimento anterior para recuperar C a partir de $\tilde{C}$, considerando o desague do pote 2 no pote 3. Para isto acontecer, a quantidade $p_3 - \tilde{p_3} > 0$ teria que ser separada da quantidade $p_2 > 0$. Porém, como observado anteriormente, não se pode utilizar nenhum equipamento que faça medições de volumes, além dos próprios potes.

Teorema 2.1 *Partindo de uma configuração* IC_0 *e executando-se um único movimento obtém-se uma configuração* IC_1.

Demonstração: Seja IC_0: $p_3 = 0,\ p_2 = c_2,\ p_1 = c_1$. Utilizando-se um movimento entre dois potes, uma das duas configurações é obtida: $p_3 = c_2,\ p_2 = 0,\ p_1 = c_1$ (desague do pote 2 no pote 3) ou IC_1: $p_3 = c_1,\ p_2 = c_2,\ p_1 = 0$ (desague do pote 1 no pote 3); que são configurações iniciais de passo 1. Estas configurações IC_1 são obtidas, na mesma ordem anterior, quando se considera IC_0, $p_3 = c_3,\ p_2 = 0,\ p_1 = 0$, com os respectivos desagues: do pote 3 no pote 1 e do

 © 2015 SBMAC

pote 3 no pote 2. $\square$

Observação: Com o intuito de se obter a solução com uma quantidade mínima de movimentos, deve-se evitar que a configuração do tipo IC_0 se repita ao longo do processo de desague entre os potes. Porque, de acordo com o Teorema anterior (2.1), partindo de uma configuração IC_0 e executando-se um único movimento obtém-se uma configuração IC_1, que é uma configuração correspondente ao início do processo.

Teorema 2.2 *Considere três números inteiros positivos, c_3, c_2 e c_1 tais que $c_3 > c_2 > c_1$; $c_3 = c_2 + c_1$; $mdc(c_i, c_j) = 1$, $i \neq j$, $i, j \in \{1, 2, 3\}$ e c_3 é par. Então $c_1 < \frac{c_3}{2} < c_2$.*

Demonstração: Note que $c_1 \neq \frac{c_3}{2}$ e $c_1 \neq \frac{c_3}{2}$, pois $mdc(c_3, c_1) = 1 = mdc(c_3, c_2)$. Observe que se $c_1 > \frac{c_3}{2}$ então $c_2 < \frac{c_3}{2}$; senão $c_1 + c_2 > c_3$. Porém, $c_1 > \frac{c_3}{2} > c_2$ não pode ocorrer já que, por hipótese, $c_3 > c_2 > c_1$. Portanto, $c_1 < \frac{c_3}{2} < c_2$. $\square$

Utilizando o resultado anterior (Teorema 2.2) pode-se demonstrar o seguinte

Teorema 2.3 *Se $p_3 = \frac{c_3}{2}$ em uma configuração C: p_3, p_2, p_1, então $p_2 = \frac{c_3}{2}$ e $p_1 = 0$ ou $p_2 = c_2 - \frac{c_3}{2} = \frac{c_3}{2} - c_1$ e $p_1 = c_1$.*

Demonstração: De acordo com a Definição 2.2, para se obter a configuração C, com $p_3 = \frac{c_3}{2}$, então, com relação aos dois potes envolvidos no movimento de desague, um dos potes deve ficar vazio ou um dos potes deve ficar completamente cheio. Pelo Teorema anterior (2.2), $c_1 < \frac{c_3}{2} < c_2$. Portanto, o pote 2 não pode estar cheio. Assim, podem ocorrer duas situações: ou o pote 1 está vazio ou está cheio, as quais conduzem às configurações propostas no enunciado do teorema. Lembre-se de que $c_3 = p_3 + p_2 + p_1$ e $c_3 = c_2 + c_1$. $\square$

Este resultado nos diz que existem apenas duas soluções possíveis para o problema dos potes, considerando-se o procedimento descrito na Definição 2.2.

3 Problema Primal e Problema Dual

Sejam c_3, c_2 e c_1 como no Teorema 2.2. O Problema Primal (PP) será aquele que tem as configurações de passo zero e de passo um dadas, respectivamente, por IC_0: $p_3 = c_3,\ p_2 = 0,\ p_1 = 0$ e IC_1: $p_3 = c_1,\ p_2 = c_2,\ p_1 = 0$.

As configurações do Problema Dual, C^*: $p_3^*,\ p_2^*,\ p_1^*$, são obtidas das configurações do Problema Primal, C: $p_3,\ p_2,\ p_1$, de modo que $p_3^* = c_3 - p_3$, $p_2^* = c_2 - p_2$ e $p_1^* = c_1 - p_1$.

A solução do PP será obtida em $c_3 - 1$ movimentos, partindo-se de IC_1 e chegando-se à configuração final, $p_3,\ p_2,\ p_1$, tal que $p_3 = p_2 = \frac{c_3}{2}$ e $p_1 = 0$. O algoritmo que fornece tal solução leva em consideração uma sequência de etapas. A primeira etapa parte de IC_0 e considera os seguintes movimentos: (i) desague do pote 3 no pote 2; (ii) desague do pote 2 no pote 1; (iii) desague do pote 1 no pote 3, até que o pote 2 fique vazio. Cada vez que o pote 2 se esvazia, inicializa-se uma nova etapa. Nesta nova etapa, a mesma sequência de movimentos descrita anteriormente é considerada.

O código abaixo foi elaborado no *software* OCTAVE e considera o caso em que $c_3 = 94$; $c_2 = 73$ e $c_1 = 21$. A Matriz MCap armazena todas as configurações obtidas ao longo do processo de divisão do volume de água entre os potes.

 © 2015 SBMAC

```
n=c3;
cont = 1;
MCap(cont,1)= c3;
MCap(cont,2) = 0;
MCap(cont,3) = 0;

for (cont = 2:n)
  MCap(cont,1)= 0;
  MCap(cont,2) = 0;
  MCap(cont,3) = 0;
end

cont= 2;
p3 = c3-c2;
MCap(cont,1) = p3;
p2=c2;
MCap(cont,2) = p2;
p1=0;
MCap(cont,3) = p1;

while (cont < n)
   if(p2 - c1 + p1 > 0)
      cont = cont+1;
      MCap(cont,1) = p3;
      p2 = p2 - (c1-p1);
      MCap(cont,2) = p2;
      p1 = c1;
      MCap(cont,3) = p1;
      cont = cont+1;
      p3 = p3+c1;
      p1 =0;
      MCap(cont,1) = p3;
      MCap(cont,2) = p2;
      MCap(cont,3) = p1;

   else
      cont = cont+1;
      p1 = p2;
      p2 = 0;
      MCap(cont,1) = p3;
      MCap(cont,2) = p2;
      MCap(cont,3) = p1;
      cont = cont+1;
      p3 = p3-c2;
      p2 = c2;
      MCap(cont,1) = p3;
      MCap(cont,2) = p2;
      MCap(cont,3) = p1;

   end
end
```

DOI: 10.5540/03.2015.003.01.0488

© 2015 SBMAC

A demonstração de que o algoritmo sempre apresenta a reposta correta não será apresentada neste trabalho. Porém, as observações dadas a seguir são importantes para este fim.

Observação 1. Pelo algoritmo de Euclides (veja Coutinho, [1], 2009), existem Q e R, números naturais, tais que $c_3 = Qc_1 + R$, com $R < c_1$. Considerando-se as mesmas hipóteses do Teorema 2.2, tem-se que $c_2 = (Q-1)c_1 + R$ e $Q > 1$.

Observação 2. Sendo c_3 par e c_1 ímpar, é fácil concluir da observação anterior que R será par (ímpar) se, e somente se, Q for par (ímpar).

Observação 3. Se R for par então $\frac{c_3}{2} = \frac{Q}{2}c_1 + \frac{R}{2}$; ou seja, $\frac{c_3}{2} \equiv \frac{R}{2}\ mod(c_1)$. Se R for ímpar então $\frac{c_3}{2} = \frac{Q-1}{2}c_1 + \frac{c_1+R}{2}$; ou seja, $\frac{c_3}{2} \equiv \frac{R+c_1}{2}\ mod(c_1)$.

Observação 4. Como mencionado anteriormente, na construção do algoritmo, cada vez que o pote 2 se esvazia, o código inicia uma nova sequência de movimentos. Seja a configuração CF, $p_3,\ p_2 = 0,\ p_1 = VF$, correspondente ao final de uma etapa, quando o pote 2 fica vazio. Assim, no início da nova etapa, a primeira configuração será CI: $VI = p_3 - c_2,\ \tilde{p_2} = c_2,\ \tilde{p_1} = VF$. Como $p_3 + VF = c_3$ então $VI = p_3 - c_2 = c_1 - VF$. Desta forma, quando R for par, $VI = c_1 - VF = \frac{R}{2}$ se, e somente se, $VF = c_1 - \frac{R}{2}$. Quando R for ímpar, $VI = c_1 - VF = \frac{c_1+R}{2}$ se, e somente se, $VF = \frac{c_1-R}{2}$.

É fácil perceber que existe uma relação importante entre as configurações e os valores de R e de Q. Para ilustrar este fato, veja a seguir as configurações correspondentes ao exemplo simulado pelo código computacional.

94	00	00
21	73	00
21	52	21
42	52	00
42	31	21
63	31	00
63	10	21
84	10	00
84	00	10

11	73	10
11	62	21
32	62	00
32	41	21
53	41	00
53	20	21
74	20	00
74	00	20

01	73	20
01	72	21
22	72	00
22	51	21
43	51	00
43	30	21
64	30	00
64	09	21
85	09	00
85	00	09

DOI: 10.5540/03.2015.003.01.0488 © 2015 SBMAC

12 73 09
12 61 21
33 61 00
33 40 21
54 40 00
54 19 21
75 19 00
75 00 19

02 73 19
02 71 21
23 71 00
23 50 21
44 50 00
44 29 21
65 29 00
65 08 21
86 08 00
86 00 08

13 73 08
13 60 21
34 60 00
34 39 21
55 39 00
55 18 21
76 18 00
76 00 18

03 73 18
03 70 21
24 70 00
24 49 21
45 49 00
45 28 21
66 28 00
66 07 21
87 07 00
87 00 07

14 73 07
14 59 21
35 59 00
35 38 21
56 38 00
56 17 21
77 17 00
77 00 17

DOI: 10.5540/03.2015.003.01.0488

© 2015 SBMAC

04 73 17
04 69 21
25 69 00
25 48 21
46 48 00
46 27 21
67 27 00
67 06 21
88 06 00
88 00 06

15 73 06
15 58 21
36 58 00
36 37 21
57 37 00
57 16 21
78 16 00
78 00 16

05 73 16
05 68 21
26 68 00
26 47 21
47 47 00

4 Considerações Finais

Eu acredito que o problema apresentado neste trabalho é relevante para o ensino de matemática porque ele tem uma característica importante: possui uma solução construtiva; ou seja, a solução pode ser obtida através de experimentos simples, sem a necessidade de argumentos matemáticos abstratos. Isto é ótimo para o ensino de matemática, porque os alunos têm a possibilidade de participar ativamente das aulas fazendo os seus próprios experimentos para obter a solução do problema.

Os professores de qualquer nível do ensino, do fundamental ao superior, podem utilizar o problema aqui proposto para introduzir a informática e as técnicas dos algoritmos computacionais no ensino de matemática. Estas são as possibilidades que se abrem para a educação do século 21.

Referências

[1] S. C. Coutinho, Número Inteiro e Criptografia RSA, Coleção Matemática e Aplicações, IMPA, Rio deJaneiro, 226 p., 2009.

DOI: 10.5540/03.2015.003.01.0488

© 2015 SBMAC