Trabalho Prático - Matemática Discreta

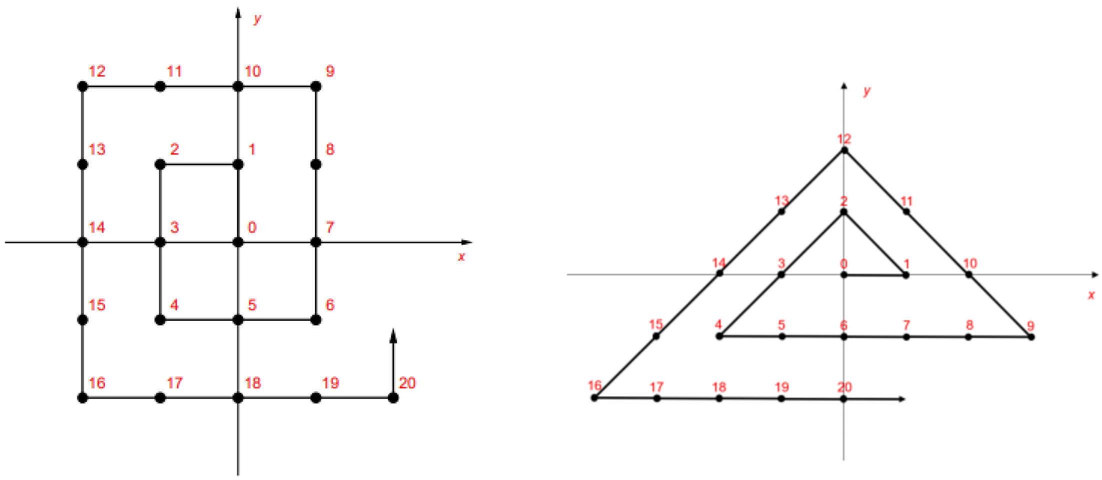
João Victor Taufner Pereira - 2017098315

[1]Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte - MG - Brasil

1. Introdução

O problema proposto nesse trabalho consiste em gerar programas que representem duas espirais, uma quadrada e uma triangular. Essas espirais são compostas por uma sequência de pontos ordenados de coordenadas x e y, e suas formas podem ser vistas nas imagens ao fim da seção.

Dessa forma, serão feitos dois programas, uma parada cada espiral, e cada um deles deve ser capaz de calcular as coordenadas de um ponto $n \geq 0$ qualquer. Para cada uma das espirais, foram dados vinte pontos iniciais e suas respectivas coordenadas.



(a) Espiral quadrada (b) Espiral triangular

Figure 1. Espirais a serem computadas

2. Implementação e Modelagem

A abordagem utilizada para a resolução do problema foi semelhante nos dois casos: Para se atingir um ponto n, foram necessários n movimentos. Isso permite que seja possivel representar um ponto qualquer dos espirais como $n = a + b$, em que a é um ponto cujas coordenadas são facilmente definidas pelo caráter cíclico dos movimentos de cada espiral e b é o excedente ou o número de movimentos que faltam para se ir de a a n. Esses b movimentos podem ser facilmente determinados para ambos espirais, como será visto ao longo da seção.

2.1. Entradas e Saídas do programa

As entradas e saídas do programa são as padrão do sistema (stdin e stdout).

- **Entrada do programa:** A entrada consiste em um único inteiro $n \geq 0$, que representa um ponto da espiral quadrada ou da triangular.

- **Saída do programa:** A saída dos programas das espirais serão as coordenadas *(x,y)* do ponto n fornecida na entrada. Essas coordenadas dependerão de qual espiral o programa executado representa.

2.2. Modelagem do problema e raciocínio utilizado

Como anteriormente mencionado, a resolução do problema consistiu em escrever o ponto a ser determinado como $n = a+b$ e, a partir disso, estudar as coordenadas e os movimentos necessários para se chegar nesses dois pontos.

Inicialmente, deve-se definir o que é um movimento. Um movimento é o deslocamento de um ponto $n \geq 0$ para o $n + 1$, alterando a coordenada em uma unidade para x ou y (ou ambos). O ponto inicial para as duas espirais é a origem, e para um ponto n qualquer, foram executados n movimentos.

A partir dessa ideia, o raciocínio chave para a resolução do problema foi desenvolvido: escrever os movimentos para se chegar a um ponto como duas somas de inteiros. O motivo de serem duas somas se deve ao caráter cíclico dos movimentos nas espirais. Por exemplo, temos os seguintes movimentos para a espiral quadrada, com i tendo valor inicial 1:

- i movimentos para cima seguidos de i movimentos para esquerda
- i movimentos para baixo seguidos de $i + 1$ movimentos para direita
- O valor de i é incrementado em 1 e volta-se ao primeiro passo

A vantagem de se enxergar os movimentos dessa forma é que o ponto alcançado ao final de cada passo possui coordenadas facilmente determinadas (como será mostrado individualmente para cada espiral). Nesse cenário, a será o valor mais próximo e menor que n que se pode atingir através da execução de um dos passos acima. Tendo-se por exemplo $n = 10$, a será igual a 6, uma vez que são feitos dois movimentos na primeira execução e quatro na segunda(totalizando seis). Caso seja feita a execução de mais um passo, o número de movimentos seria 12, ultrapassando n. Com a determinado, b será simplesmente o número de movimentos necessários para se ir de a a n ($b = 4$ para este exemplo).

Será necessário, portanto, encontrar uma forma de determinar a para um n qualquer. Dado o ponto de inicial na origem, deseja-se seguir os passos supracitados para se chegar a n. Os movimentos podem ser escritos, para $x \in \mathbb{N}$ como:

$$1 + 1 + 2 + 2 + 3 + 3 + ... + x + x = 2 * \sum_{i=1}^{x} i = x(x + 1) = n$$

Calculando a raíz positiva da equação de segundo grau acima, é possível determinar qual era o último inteiro da soma de movimentos que resultou em n. Chamaremos esse resultado de *salto*. Para determinar a, basta-se substituir o valor do salto em $x(x + 1)$. É importante citar que a raíz da equação nem sempre será natural. Nesses casos, basta-se tomar o piso da raíz.

Esse raciocínio será utilizado da mesma maneira tanto para o algorítmo da espiral quadrada quanto para o da triangular. A diferença entre as espirais encontra-se na determinação das coordenadas de a e dos posteriores b movimentos.

2.3. Implementação

Os algorítmos geradores das duas espirais possuem as mesmas variáveis.

- ***input***: armazena o valor de entrada do programa.
- ***x* e *y***: representam as coordenadas da saída.
- ***resolve***: representa a raíz positiva da equação de segundo grau anteriormente mencionada.
- ***limite***: representa a.
- ***falta***: representa b.
- ***salto***: armazena o valor de resolve acrescido em um.

A relevância da variável salto se justifica pela necessidade de saber quantos movimentos serão feitos no próximo passo, depois que se atinge a. Por exemplo, para $n = 10$, $a = 6$ e *salto* $= 3$. Com esses valores, será possível determinar as coordenadas de qualquer ponto das espirais.

2.3.1. Espiral Quadrada

Para o caso da espiral quadrada, existem duas possibilidades para a coordenada de a, advindas da descrição dos movimentos feita anteriormente.

- Caso a seja atingido por um passo com movimentos pares(valor guardado na variável *resolve*), a coordenada x de a será igual ao valor do salto divido por dois. Isso é facilmente percebido pois a cada dois passos, o valor de x é acrescido em um. Além disso, o valor da coordenada y será o salto divido por dois, multiplicado por -1. O motivo é análogo, mas o valor de y é decrescido em um a cada dois passos. Para valores encontrados não inteiros, basta-se tomar o piso.
- Caso a seja atingido por um passo com movimentos impares, a coordenada x de a será o salto divido por dois, multiplicado por -1. Já y será o valor do salto divido por dois. Esses valores decorrem diretamente dos resultados mostrados para o caso do movimento com passos pares.

Com as coordenadas de a determinadas, basta se adicionar os b movimentos restantes para obter a coordenada de n. Esses movimentos são triviais: são os primeiros b movimentos do passo posterior ao que atingiu a. Ou seja, se atingimos a por i movimentos para cima e i movimentos para a esquerda, o próximo passo será de $i + 1$ movimentos para baixo seguidos de $i + 1$ movimentos para direita.

Para saber exatamente quantos devem ser dados em cada direção, basta comparar os valores de b e i. Se $b \geq i$, então fazemos i movimentos na primeira direção do passo e $b - i$ na segunda direção. Caso $b < i$, basta fazer b movimentos na primeira direção do passo.

Para ilustrar a execução do algorítmo, considere a determinação das coordenadas de $n = 17$. Nesse caso, o valor de a obtido pelas equações será 12(e por consequência, $b = 5$), e o salto será 4. Como o valor do piso da raíz da equação é 3, sabemos que 12 foi atingido por um passo de movimentos ímpares. Portanto as coordenadas de a são $(-2, 2)$. Como *salto* $= 4$, sabemos que no próximo passo constariam quatro movimentos para baixo seguidos de quatro movimentos para a direita. Então, como $b = 5 > 4$, faremos 4 movimentos para baixo e $5 - 4$ moviemntos para a direita, resultando nas coordenadas $(-1, -2)$ do ponto n.

2.3.2. Espiral Triangular

A descrição dos movimentos da espiral triangular se dá da seguinte forma:

- i movimentos para direita seguidos de i movimentos no sentido noroeste(cada um diminui em um o valor da coordenada x e aumenta em um o valor da coordenada y).
- $i+1$ movimentos no sentido sudoeste(diminuindo em um ambas as coordenadas) seguidos de $i+1$ movimentos para a direita.
- O valor de i é incrementado em 1 e volta-se ao primeiro passo

Os passos foram escolhidos dessa maneira para facilitar o cálculo das coordenadas de a, de forma que a coordenada x sempre seja 0.

O funcionamento do algorítmo é análogo ao utilizado para espiral quadrada, mas como os passos são diferentes, algumas alterações precisam ser feitas. Novamente, temos duas possibilidades para as coordenadas de a.

- Caso a seja atingido por um passo de movimentos pares, a coordenada y será o valor do salto dividido por dois, multiplicado por -1. Isso se deve ao fato de que a cada dois passos, o valor de y é reduzido em um. Para valores não inteiros, basta-se tomar o piso.
- Caso a seja atingido por um passo de movimentos ímpares, a coordenada y será o valor do salto divido por dois. Esse resultado decorre diretamente do anterior, uma vez que basta tomar um caso de movimentos pares e ”desfazer” o último passo.

Semelhante ao caso da espiral quadrada, basta agora executar os b movimentos restantes a partir da comparação com o valor i. Como a explicação de como essa etapa funciona já foi feita anteriormente, ela será demonstrada por meio de um exemplo. Considere a execução do algorítmo da espiral triangular para $n = 26$. Nesse caso, o valor obtido de a será $20(b = 6)$ e o *salto* será 5 (piso da positiva raíz da equação é 4). Como 4 é par, temos que a é atingido por um passo de movimentos pares. Portanto suas coordenadas são $x = 0$ e y = $-1 * \lfloor salto/2 \rfloor = -2$. Logo a está em $(0, -2)$. Como *salto* $= 5$, sabemos que no próximo passo constariam cinco movimentos para a direita seguidos de cinco movimentos no sentido noroeste. Como $b = 6 > 5$, serão executados cinco movimentos para direita e $6 - 5 = 1$ movimentos no sentido noroeste, resultando nas coordenadas $(4, -1)$ para o ponto n.

3. Análise de Complexidade

Tanto para o algorítmo que representa a espiral quadrada quanto para o algorítmo que representa a triangular, o mesmo número de operações é executado independente do valor passado na entrada. Dessa forma, a complexidade de ambos os algorítmos é da forma $\Theta(1)$.

4. Conclusões

A partir da implementação do trabalho foi possível colocar em prática os conhecimentos adquiridos na disciplina a respeito de notações assintótica de algorítmos, além de encontrar uma solução para o problema proposto cujo tempo de execução independe do valor de entrada.