Matemática

Funções

Prof. Edson Alves Faculdade UnB Gama

Produtos Cartesianos

- ullet Sejam A e B dois conjuntos
- ullet O **produto cartesiano** de A por B é dado por

$$A imes B=\{(a,b)\mid a\in A,b\in B\}$$

- ullet Em outras palavras, é o conjunto de todos os possíveis pares ordenados cujo primeiro elemento pertence a A e o segundo pertence a B
- \bullet Se A tem n elementos e B tem m elementos, o produto cartesiano terá nm elementos distintos
- ullet Observe que se A
 eq B então A imes B
 eq B imes A

Relações

- ullet Sejam A e B dois conjuntos
- ullet Dizemos que R é uma **relação** de A em B se $R\subset A imes B$, isto é, se R é um subconjunto do produto cartesiano de A por B
- ullet Se |A|=n e |B|=m, existem 2^{nm} relações de A em B
- ullet Se $(a,b)\in R$, dizemos que a se relaciona com b
- ullet Observe que $(a,b)\in R$ não implica $(b,a)\in R^1$

Funções

Uma relação $f\in A imes B$ é uma **função** de A em B (e escrevemos f:A o B) se os dois critérios abaixo forem atendidos:

- 1. todo elemento a de A se relaciona com algum elemento b de B;
- 2. cada elemento a de A está relacionado com um único elemento b de B.

Injeção, sobrejeção e bijeção

- ullet Uma função f:A o B é dita **injetora** se f(a)=f(b) implica em a=b, isto é, cada elemento do conjunto B está relacionado com um único elemento do conjunto A
- f é dita **sobrejetora** se, para qualquer elemento $b\in B$, existe um elemento $a\in A$ tal que f(a)=b, ou seja, todos os elementos de B estão relacionado a ao menos um elemento de A
- Uma função que é, ao mesmo tempo, injetora e sobrejetora é dita bijetora

Função inversa

- A classificação de uma função como injetora ou sobrejetora está relacionada diretamente aos dois critérios da definição de funções
- ullet Considere uma função f:A o B e seja $R\subset B imes A$ uma relação de B em A dada por

$$R=\{(b,a)\mid a\in A,b\in B,f(a)=b\}$$

- ullet Se a relação R atender ao primeiro critério, então a função f é sobrejetora; se atender o segundo critério, f é injetora
- ullet Se a relação R atende a ambos critérios, R é uma função, denominada função **inversa** de f

Funções invertíveis

- ullet Uma função f é invertível se for bijetora
- ullet A função inversa de f:A imes B, se existir, é grafada como $f^{-1}:B imes A$
- ullet Se for invertível, f estabelece uma relação um-a-um entre os elementos de A e B
- ullet Se A e B forem conjuntos finitos, então ambos terão o mesmo número de elementos

Variáveis independentes

- Na notação y=f(x), x é a variável **independente** e y é a variável **dependente**: dizemos que y é função de x, ou que y depende de x
- ullet Isto significa que, conhecido o valor de x, é possível determinar o valor de y
- Uma variável pode ser dependente de mais de uma variável
- ullet Por exemplo, área A de um retângulo depende dos valores das medidas da base b e da altura h do retângulo, ou seja, A=A(b,h)

Zeros de funções

- ullet Seja f:A o B, onde 0 (zero) pertence a B
- ullet Dizemos que $x\in A$ é um **zero** de f se f(x)=0
- Uma função pode não ter, ter finitos ou infinitos zeros
- Exemplos:
 - \circ a função f(x)=1/x não tem zeros nos reais
 - \circ o Teorema Fundamental da Álgebra diz que todo polinômio de grau n tem n raízes complexas
 - \circ a função $f(x) = \operatorname{sen}(x)$ tem infinitos zeros: qualquer múltiplo de 2π

Método da bisseção

- ullet Seja $f:\mathbb{R} o\mathbb{R}$ é uma função contínua em um intervalo I dos reais, isto é, para qualquer elemento $a\in I$, o limite de f(x) quanto x tende a a existe e é igual a f(a)
- ullet Suponha que existam dois valores $a,b\in I$ tais que f(a)f(b)<0, isto u, f(a) e u
- ullet Nestas condições, o Teorema de Valor Intermediário garante que existe ao menos um valor $c\in(a,b)$ tal que f(c)=0
- ullet O método da bisseção consiste em aproximar o valor de c por meio de uma busca binária

```
// Assuma que a função f(double) esteja definida, que a < b e que f(a)*f(b) < 0
// eps é a tolerância de erro
double root(double a, double b, double eps)
    while (fabs(a - b) > eps)
        auto c = (a + b)/2;
        auto y = f(c);
        // c é uma boa aproximação para o zero
        if (fabs(y) < eps)</pre>
            return c;
        // Determina em qual dos intervalos ( (a,c) ou (c,b) ) está o zero
        f(a)*y < 0? b = c : a = c;
    return (a + b)/2;
```

Convergência

- Por conta de possíveis erros de precisão, o método da bisseção pode não convergir ou não melhorar sua precisão após um determinado número de iterações
- ullet Implementações alternativas usam um número N de passos prédeterminado como critério de parada
- Há outros métodos com melhor convergência, como o método de Newton
- Ainda o método da bisseção é notável por sua simplicidade e aplicabilidade

```
// Assuma que a função f(double) esteja definida, a < b e que f(a)*f(b) < 0
// N é o número de iterações do algoritmo
double root(double a, double b, int N)
    while (N--)
       double c = (a + b)/2;
        // Determina em qual dos intervalos ( (a,c) ou (c,b) ) está o zero
        f(a)*f(c) < 0 ? b = c : a = c;
    return (a + b)/2;
```

Problemas

- AtCoder
 - 1. ABC 043B Be Together
- Codeforces
 - 1. 486A Calculating Function
 - 2. <u>1036A Function Height</u>
- OJ
 - 1. 371 Ackermann Functions
 - 2. <u>10431 Solve It</u>

Referências

1. Wikipédia. <u>Bissection Method</u>. Acesso em 15 de agosto de 2017.