Técnicas de programação para Games
Aula02
Bases Matemáticas

Professor Mestre: Adilson Lopes Khouri

14 de outubro de 2018

Sumário

Recursão

Cronograma

Aula	**Conteúdo**
12/04/2018	XP e banco de dados
17/04/2018	Introdução de estruturas de dados
24/04/2018	Arrays / Matrizes e Ordenação
26/04/2018	Recursão
03/05/2018	Lista Ligada
08/05/2018	Pilha, Fila
10/05/2018	Hash
15/05/2018	Árvore Binária
17/05/2018	Heap
22/05/2018	Grafos
24/05/2018	Prova

Recursão

- O que é recursão? Uma função que chama ela mesma diversas vezes até convergir em um caso básico do problema, de solução simples. Após atingir esse caso, ela combina todos os resultados intermediários até retornar a solução do problema.
- É um paradigma computacional muito poderoso, permite resolver problemas de forma simples. Programas recursivos tem uma tendência de serem menores que sua abordagens iterativas.
- Está relacionada com conceito de indução matemática.

Indução matemática

- Indução matemática é uma técnica matemática para provar asserções sobre números naturais
- Dado um teorema T com parâmetro N, que é um número natural. Precisamos provar que T é válido.
- Usando indução matemática fazemos o seguinte procedimento:
- Provamos T para $n = 1$ (chamado caso base)
- Para todo T onde $N > 1$, se T é válido para $n - 1$ então é válido para N (passo indutivo)

Indução matemática

- Por que isso funciona? Por que apenas essas duas condições são suficientes?
- Simples, as condições 1 e 2 implicam que T é válido para $n = 2 \ldots$
- Pela condição 2, se for válido para 2 será para 3... e assim sucessivamente.

Indução matemática

- Um exemplo matemático, a soma dos n primeiros números naturais n, pode ser escrita como: S(n) = 1+2+...+n. Sabemos que isso é a soma de uma PA com r = 1, como provar que a soma desses números é dada pela fórmula fechada: S(n) = n*(n+1)/2 ?
- Podemos usar indução!
- Passo base: n = 1 S(1) = 1 (trivial)
- Passo Indutivo:
-

$$S(n) = S(n-1) + n \Leftrightarrow \quad (1)$$

$$((n-1)*(n-1+1))/2 + n \Leftrightarrow \quad (2)$$

$$n*(n+1)/2 \quad (3)$$

- A partir da hipótese indutiva com parâmetro n-1 (fórmula fechada) chegamos em n provando por indução que funciona

Indução matemática - Exercício

- Como multiplicar dois números sem usar o operador de multiplicação? Programar a versão iterativa e a recursiva...

Indução matemática - Exercício

- A sequência de Fibonacci pode ser escrita como:
- $F(n)$ se $n <= 1$ então retorne n
- caso contrário$f(n-1)+f(n-2)$
- Vamos programar esse algoritmo....

Indução matemática

- Uma analogia usada para compreender recursão são as bonecas russas matrioskas

Figura: Bonecas Matrioskas

Indução matemática "Forte"

- Indução forte é uma estratégia parecida com indução fraca mas que origina algoritmos baseados em divisão e conquista.
- Novamente devemos provar o caso base:
- $N = 1$
- Mas agora, ao invés de provar para $n - 1$ vamos provar que a hipótese é válida para qualquer $k < n$

Indução matemática "Forte"

- Citar algoritmo de exponenciação por indução forte
- $N = 0$ então $exp(a, 0) = 1$ (caso base)
- Passo indutivo, sei calcular $exp(a, piso(n/2))$
- O cálculo seria: se $n\%2 = 0$ então $exp(a, piso(n/2))^2$
- se $n\%2 = 1$ então $exp(a, piso(n/2))^2$
- Explicar no quadro e programar!

Tribonacci

- A sequencia de Tribonacci é muito parecida com a série de Fibonacci. A diferença é que cada elemento é a soma dos três anteriores.
- Mostrar no quadro e programar com os alunos

Dúvidas...

Alguma dúvida?

Contato

- E-mail: 0800*dirso@gmail.com* (alunos SENAC)
- E-mail: *adilson.khouri.usp@gmail.com*
- Phone: $+55119444-26191$
- Linkedin
- Lattes
- GitHub

Referências

[1] A. V. Aho, J. E. Hopcroft, and J. Ullman, *Data Structures and Algorithms*, 1st ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1983.

[2] K. Beck and C. Andres, *Extreme Programming Explained: Embrace Change (2Nd Edition)*. Addison-Wesley Professional, 2004.

[3] Beck, *Test Driven Development: By Example*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2002.

[4] M. Fowler, *Refactoring: Improving the Design of Existing Code*. Boston, MA, USA: Addison-Wesley, 1999.

[5] Geeks. (2018) A computer science portal for geeks. [Online]. Available: https://www.geeksforgeeks.org