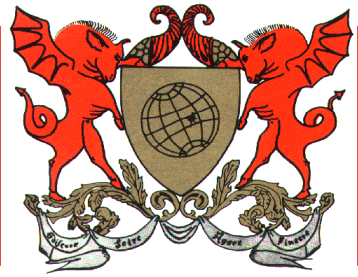

Universidade Federal de Viçosa
Campus de Florestal

Algoritmos e Estruturas de Dados I (CCF 211)

Shellsort
(Cap - Ziviani)

Profa.Thais R. M. Braga Silva
<thais.braga@ufv.br>

ShellSort

- **Proposto por Shell em 1959.**
- **É uma extensão do algoritmo de ordenação por inserção.**
- **Problema com o algoritmo de ordenação por inserção:**
  - **Troca itens adjacentes para determinar o ponto de inserção.**
  - **São efetuadas n - 1 comparações e movimentações quando o menor item está na posição mais à direita no vetor.**
- **O método de Shell contorna este problema permitindo trocas de registros distantes um do outro.**

ShellSort

- **Os itens separados de *h* posições são rearranjados.**
- **Todo *h*-ésimo item leva a uma sequência ordenada.**
- **Tal sequência é dita estar *h*-ordenada.**

Exemplo

O R D E N A

h = 4

h = 2

h = 1

ShellSort

- **Exemplo de utilização:**

	1	2	3	4	5	6
Chaves iniciais:	*O*	*R*	*D*	*E*	*N*	*A*
h = 4	*N*	*A*	*D*	*E*	*O*	*R*
h = 2	*D*	*A*	*N*	*E*	*O*	*R*
h = 1	*A*	*D*	*E*	*N*	*O*	*R*

- **Quando *h* = *1*, Shellsort corresponde ao algoritmo de inserção.**

ShellSort

- **Como escolher o valor de *h*:**
  - **Sequência para h:**

    h(s) = 1, para s = 1

    h(s) = 3h(s - 1) + 1, para s > 1

ShellSort

- **Como escolher o valor de *h*:**
  - **Sequência para h:**

    h(s) = 1, para s = 1

    h(s) = 3h(s - 1) + 1, para s > 1

  - **A sequência para h corresponde a 1, 4, 13, 40, 121, 364, 1.093, 3.280, ...**
  - **Knuth (1973, p. 95) mostrou experimentalmente que esta sequência é difícil de ser batida por mais de 20% em eficiência.**

ShellSort

```
void Shellsort (Item* A, int n){
  int i, j;
  int h = 1;
  Item aux;

  do h = h * 3 + 1; while (h < n);
  do
  {
    h = h/3;
    for( i = h ; i < n ; i++ )
    {
      aux = A[i]; j = i;
      while (A[j – h].Chave > aux.Chave)
      {
        A[j] = A[j – h]; j -= h;
        if (j < h) break;
      }
      A[j] = aux;
    }
  } while (h != 1);
}
```

ShellSort

- **Análise**
  - **A razão da eficiência do algoritmo ainda não é conhecida.**
  - **Ninguém ainda foi capaz de analisar o algoritmo.**
  - **A sua análise contém alguns problemas matemáticos muito difíceis.**
  - **A começar pela própria sequência de incrementos.**
  - **O que se sabe é que cada incremento não deve ser múltiplo do anterior.**

ShellSort

- **Análise**
  - **Conjecturas referente ao número de comparações para a sequência de Knuth:**

Conjectura 1 : $C(n) = O(n^{1,25})$

Conjectura 2 : $C(n) = O(n\ (\ln n)^2\ )$

ShellSort

- **Vantagens:**
  - **Shellsort é uma ótima opção para arquivos de tamanho moderado.**
  - **Sua implementação é simples e requer uma quantidade de código pequena.**

- **Desvantagens:**
  - **O tempo de execução do algoritmo é sensível à ordem inicial do arquivo.**
  - **O método não é estável.**