

Classificação e Pesquisa de Dados

Aula 04 e 05
Classes de Métodos de Classificação;
Classificação de dados por Inserção Direta
UFRGS INF01124

1. Classificação por inserção

- Inserção Direta
- Inserção Direta com Busca Binária
- Método dos Incrementos Decrescentes (*ShellSort*)

Instituto de Informática - UFRGS

Métodos de Classificação de Dados

1. **Classificação por Inserção**
2. Classificação por Trocas
3. Classificação por Seleção
4. Classificação por Distribuição de Chaves
5. Classificação por Intercalação
6. Classificação por Cálculo de Endereços

Instituto de Informática - UFRGS

Inserção Direta: exercício

- ◆ Utilize o algoritmo de classificação por inserção direta para classificar os seguintes arrays:

◆ $C = \langle 6, 3, 5, 8, 2, 4 \rangle$

◆ $C = \langle 2, 3, 4, 5, 6, 8 \rangle$

◆ $C = \langle 8, 6, 5, 4, 3, 2 \rangle$

Instituto de Informática - UFRGS

Método dos Incrementos Decrescentes (*ShellSort*)

- ◆ Dr. Donald L. Shell é o autor do *ShellSort*.
- ◆ Explora o fato de que o método da **Inserção Direta** apresenta desempenho aceitável quando:
 - ◆ O **número** de chaves a ser ordenado é **pequeno** e/ou
 - ◆ As chaves apresentam uma **ordenação parcial**
- ◆ Divide o vetor em h segmentos
 - ◆ Segmento 1: $C[1], C[h+1], C[2h+1], \dots$
 - ◆ Segmento 2: $C[2], C[h+2], C[2h+2], \dots$
 - ◆ Segmento i : $C[i], C[h+i], C[2h+i], \dots$

ShellSort

1. Ordenação é realizada dividindo o conjunto em subconjuntos
2. Os subconjuntos são constituídos por elementos separados h elementos (**incremento**).
3. Os subconjuntos são ordenados utilizando o método de inserção direta
4. Repetição dos pontos **1, 2 e 3, com diminuição do incremento**, até o conjunto estar ordenado.

Método dos Incrementos Decrescentes

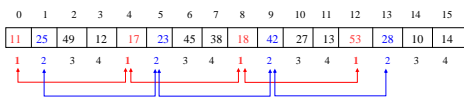
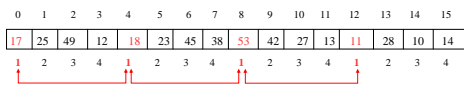
- ◆ Dado um valor inicial de h (potência de 2)
 - i. Classifica cada um dos segmentos utilizando inserção direta;
 - ii. Reduz o valor de h à **metade** e repete o passo (i) até que $h = 1$ **incrementos decrescentes**;
 - iii. Ao repetir o procedimento para $h = 1$, o vetor estará ordenado.
- ◆ Cada passo faz **classificações parciais** do vetor

Método dos Incrementos Decrescentes

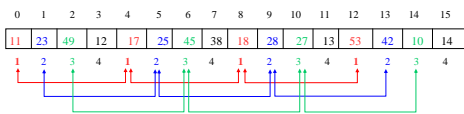
Primeiro passo: $h = 4$

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17	25	49	12	18	23	45	38	53	42	27	13	11	28	10	14
1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4

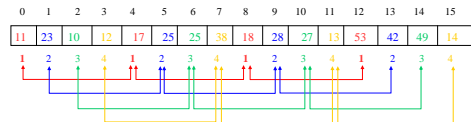
Primeiro passo: $h = 4$



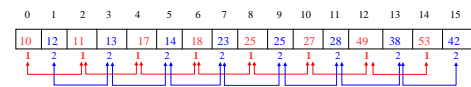
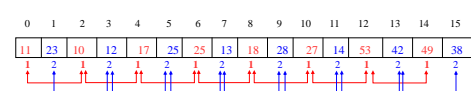
Primeiro passo: $h = 4$



Primeiro passo: $h = 4$



Primeiro passo: $h = 2$



Primeiro passo: $h = 1$



Inserção Direta ...

Sequencias para H

- Knuth (1973, p. 95)
 - Provou experimentalmente a seguinte fórmula
- A sequência para h corresponde:

para $s > 1$	$h(s) = 3^s h - 1$
para $s = 1$	$h(s) = 1$
- A sequência para h corresponde
 - 1, 4, 13, 40, 121, 364, 1093, 3280 ...

Inserção direta

```
int main()
{
    int v[] = {2, 6, 8, 4};
    int n=5;
    int i, j, chave, copia=0, comp=0;

    for(j = 1; j < n; j++)
    {
        chave = v[j];
        i = j - 1;
        while(i >= 0 && v[i] > chave)
        {
            v[i + 1] = v[i];
            i = i - 1;
        }
        v[i + 1] = chave;
    }
}
```

Método dos Incrementos Decrescentes

```
int main()
{
    int v[] = {2, 6, 8, 4};
    int n=5;
    int i, j, chave, copia=0, comp=0;

    for(j = 1; j < n; j++)
    {
        chave = v[j];
        i = j - 1;
        while(i >= 0 && v[i] > chave)
        {
            v[i + 1] = v[i];
            i = i - 1;
        }
        v[i + 1] = chave;
    }
}
```

```
void main()
{
    int i, j, k, h = 1, value, numero = 10 ;
    int vetor[] = {7, 2, 5, 6, 3, 1, 0, 8, 9, 4};

    do { h = 3 * h + 1; } while (h < numero );

    do
    {
        h /= 3;
        for ( i = h; i < numero; i++)
        {
            value = vetor [ i ];
            while ( j >= 0 && value < vetor [ j ] )
            {
                vetor [ j + h ] = vetor [ j ];
                j = j - h;
            }
            vetor [ j + h ] = value;
        }
    } while ( h > 1 );
}
```

Método dos Incrementos Decrescentes

```
void main()
{
    int i, j, k, h = 1, value, numero = 10 ;
    int vetor[] = {7, 2, 5, 6, 3, 1, 0, 8, 9, 4};

    do { h = 3 * h + 1; } while ( h < numero );

    do
    {
        h /= 3;
        for ( i = h; i < numero; i++)
        {
            value = vetor [ i ];
            j = i - h;
            while ( j >= 0 && value < vetor [ j ] )
            {
                vetor [ j + h ] = vetor [ j ];
                j = j - h;
            }
            vetor [ j + h ] = value;
        }
    } while ( h > 1 );
}
```

Método dos Incrementos Decrescentes

• Análise do Desempenho:

- A análise do desempenho do método é complexa, envolvendo problemas matemáticos difíceis, alguns deles ainda não resolvidos;
- Um dos problemas é determinar o efeito que a ordenação dos segmentos em um passo produz nos passos subsequentes
- Outros fatores determinantes do tempo de execução incluem o número de passos, número de comparações e número de trocas
- Também não se conhece a sequência de incrementos que produz o melhor resultado para grandes valores de n .