INF221 – Algoritmos y Complejidad

Clase #18
Complejidad

Horst H. von Brand

11 de octubre de 2016

1. Métodos simples de ordenamiento

Tenemos los métodos de la burbuja (listado 1), selección, (listado 2), e inserción (listado 3). Es fácil ver que todos ellos tienen tiempo de ejecución $O(n^2)$. El método de selección tiene mejor caso $O(n^2)$, los otros dos tienen mejor caso $O(n)$.

```
void
sort(double a[], int n)
{
        int k;

        for (int i = n - 1; i; i = k) {
                k = 0;
                for (int j = 0; j < i; j++) {
                        if (a[j + 1] < a [j]) {
                                double tmp;
                                tmp = a[j]; a[j] = a[j + 1]; a[j + 1] = tmp;

                                k = j;
                        }
                }
                i = k;
        }
}
```

Listado 1: Método de la burbuja

1.1. Rendimiento de métodos simples de ordenamiento

Nos interesa obtener información más detallada que ésta sobre el rendimiento de estos algoritmos. En particular, interesa el tiempo promedio de ejecución. Para ello debemos considerar una distribución de los datos de entrada (valores repetidos, orden original del arreglo). Para simplificar, supondremos que no hay datos repetidos, y como los algoritmos únicamente comparan elementos podemos asumir que la entrada es una permutación de $1, \ldots, n$. O sea, requerimos la distribución de las permutaciones dadas al algoritmo. Esto en general es imposible de conseguir (y engorroso de tratar), así que supondremos que todas las permutaciones son igualmente probables. Esto reduce el análisis detallado a derivar propiedades promedio de las permutaciones.

Definición 1.1. Una *inversión* de la permutación π de $1, \ldots, n$ es un par de índices i, j tales que $i < j$ y $\pi(i) > \pi(j)$.

```
void
sort(double a[], int n)

{
        for(int i = 0; i < n; i++) {
                int imin = i; double min = a[i];
                for(int j = i + 1; j < n; j++)
                        if(a[j] < min) {
                                imin = j; min = a[j];
                        }
                double tmp = a[i]; a[i] = a[imin]; a[imin] = tmp;
        }
}
```

Listado 2: Método de selección

```
void
sort(double a[], int n)
{
        for (int i = 1; i < n; i++) {
                double tmp = a[i];
                int j;
                for (j = i - 1; j >= 0 && tmp < a[j]; j--)
                        a[j + 1] = a[j];
                a[j + 1] = tmp;
        }
}
```

Listado 3: Método de inserción

El número mínimo de inversiones es 0 (el arreglo ordenado no tiene inversiones), el máximo es $n(n-1)/2$ (en el arreglo ordenado de mayor a menor el elemento i participa en $i-1$ inversiones con elementos mayores, sumando para $1 \le i \le n$ se tiene el valor citado).

Es claro que en el método de la burbuja cada intercambio elimina exactamente una inversión. Vale decir, el número de asignaciones de elementos es tres veces el número de inversiones.

El método de inserción funciona esencialmente como el de la burbuja, solo que en vez de intercambiar en cada paso deja un espacio libre en la posición original, copia cada elemento una posición hacia arriba si es mayor que el elemento bajo consideración, moviendo la posición libre hacia abajo; finalmente ubica el elemento en su posición (la que queda libre después de los malabares anteriores). Esto se resume en dos asignaciones para cada elemento, y una asignación adicional para cada inversión.

1.2. Funciones generatrices cumulativas

Comparar los métodos es entonces esencialmente obtener el número medio de inversiones en las permutaciones de $1, \ldots, n$. Para ello recurrimos a nuestra técnica preferida, funciones generatrices. De manera muy similar a como contabilizamos las estructuras de un tamaño dado mediante funciones generatrices podemos representar el total de alguna característica. Dividiendo por el número de estructuras del tamaño respectivo tenemos el promedio del valor de interés.

Para precisar, consideremos una clase de objetos $\mathscr{A}$. Como siempre el número de objetos de tamaño n lo anotaremos a_n, con función generatriz:

$$\begin{aligned} A(z) &= \sum_{\alpha \in \mathscr{A}} z^{|\alpha|} && (1) \\ &= \sum_{n \ge 0} a_n z^n && (2) \end{aligned}$$

Consideremos no sólo el número de objetos, sino alguna característica, cuyo valor para el objeto α anotaremos $\chi(\alpha)$. Es natural definir la *función generatriz cumulativa*:

$$C(z) = \sum_{\alpha \in \mathscr{A}} \chi(\alpha) z^{|\alpha|} \tag{3}$$

Vale decir, los coeficientes son la suma de la medida χ para un tamaño dado:

$$[z^n]C(z) = \sum_{|\alpha|=n} \chi(\alpha) \tag{4}$$

Así tenemos el valor promedio para objetos de tamaño n:

$$\mathbb{E}_n[\chi] = \frac{[z^n]C(z)}{[z^n]A(z)} \tag{5}$$

La discusión precedente es aplicable si tenemos objetos no rotulados entre manos. Si corresponden objetos rotulados, podemos definir las respectivas funciones generatrices exponenciales:

$$\widehat{A}(z) = \sum_{\alpha \in \mathscr{A}} \frac{z^{|\alpha|}}{|\alpha|!} \tag{6}$$

$$= \sum_{n \geq 0} a_n \frac{z^n}{n!} \tag{7}$$

$$\widehat{C}(z) = \sum_{\alpha \in \mathscr{A}} \chi(\alpha) \frac{z^{|\alpha|}}{|\alpha|!} \tag{8}$$

Nuevamente, como los factoriales en los coeficientes se cancelan:

$$\mathbb{E}_n[\chi] = \frac{[z^n]\widehat{C}(z)}{[z^n]\widehat{A}(z)} \tag{9}$$

1.3. Análisis de burbuja e inserción

Nuestros objetos de interés son permutaciones, objetos rotulados. Corresponde usar funciones generatrices exponenciales.

Anotemos $\iota(\pi)$ para el número de inversiones de la permutación π, y definamos la función generatriz cumulativa:

$$I(z) = \sum_{\pi \in \mathscr{P}} \iota(\pi) \frac{z^{|\pi|}}{|\pi|!} \tag{10}$$

En particular, nos interesa el número promedio de inversiones para permutaciones de tamaño n.

Podemos describir permutaciones mediante la expresión simbólica:

$$\mathscr{P} = \mathscr{E} + \mathscr{P} \star \mathscr{Z} \tag{11}$$

Vale decir, una permutación es vacía o es una permutación combinada con un elemento adicional. Dada la permutación π construimos permutaciones de largo $|\pi|+1$ añadiendo un nuevo elemento vía la operación $\star$. Estamos creando $|\pi|+1$ nuevas permutaciones, cada una de las cuales conserva las inversiones que tiene, y agrega entre 0 y $|\pi|$ nuevas inversiones dependiendo del valor elegido como último. El total de inversiones en el conjunto de permutaciones así creado a partir de π es:

$$(|\pi|+1)\iota(\pi) + \sum_{0 \leq k \leq |\pi|} k = (|\pi|+1)\iota(\pi) + \frac{|\pi|(|\pi|+1)}{2} \tag{12}$$

Con esto tenemos la descomposición para la función generatriz cumulativa (la permutación de cero elementos no tiene inversiones):

$$I(z) = \sum_{\pi \in \mathscr{P}} \iota(\pi) \frac{z^{|\pi|}}{|\pi|!} \tag{13}$$

$$= \iota(\epsilon) + \sum_{\pi \in \mathscr{P}} \left((|\pi|+1)\iota(\pi) + \frac{|\pi|(|\pi|+1)}{2} \right) \frac{z^{|\pi|+1}}{(|\pi|+1)!} \tag{14}$$

$$= \sum_{\pi \in \mathscr{P}} \iota(\pi) \frac{z^{|\pi|+1}}{|\pi|!} + \frac{1}{2} \sum_{\pi \in \mathscr{P}} \frac{z^{|\pi|+1}}{|\pi|!} |\pi|$$

Como hay $k!$ permutaciones de tamaño k, sumando sobre tamaños resulta:

$$\begin{aligned} &= zI(z) + \frac{1}{2} z \sum_{k \geq 0} k z^k \\ &= zI(z) + \frac{z^2}{2(1-z)^2} \end{aligned} \tag{15}$$

Despejando:

$$I(z) = \frac{1}{2} \frac{z^2}{(1-z)^3} \tag{16}$$

Obtenemos el número promedio de inversiones directamente, ya que hay $n!$ permutaciones de tamaño n, y el promedio casualmente es el coeficiente de z^n en la función generatriz exponencial:

$$\mathbb{E}_n[\iota] = [z^n] I(z) \tag{17}$$

$$\begin{aligned} &= \frac{1}{2} [z^n] \frac{z^2}{(1-z)^3} \\ &= \frac{1}{2} [z^{n-2}] (1-z)^{-3} \\ &= \frac{1}{2} \binom{-3}{n-2} \\ &= \frac{1}{2} \binom{n-2+(3-1)}{3-1} \\ &= \frac{1}{2} \binom{n}{2} \\ &= \frac{n(n-1)}{4} \end{aligned} \tag{18}$$

Podemos resumir las anteriores como número de asignaciones asintóticas al ordenar n elementos en el cuadro 1. Queda claro que (salvo para gente optimista incurable) el método de inserción es mejor.

Algoritmo	**Min**	**Prom**	**Máx**
Burbuja	0	$3n^2/4$	$3n^2/2$
Inserción	$2n$	$n^2/4$	$n^2/2$

Cuadro 1: Comparación entre métodos de burbuja e inserción

1.4. Análisis de selección

El método de selección tiene sentido si queremos minimizar el número de copias (podemos simplemente copiar y comparar claves, y copiar elementos solo para ubicarlos en su lugar). En tal caso interesa fundamentalmente el número de asignaciones.

Para el método de selección el número de asignaciones está dado por el número de veces que hallamos un elemento mayor, o sea, el número de máximos de izquierda a derecha en la permutación. Si llamamos $\chi(\pi)$ al número de máximos de izquierda a derecha en la permutación π, como el último elemento de la permutación es un máximo de izquierda a derecha si es el máximo de todos ellos, usando la convención de Iverson podemos expresar el número de máximos de izquierda a derecha en la permutación resultante de $\pi \star (1)$ si se asigna el rótulo j al elemento nuevo como:

$$\chi(\pi) + [j = |\pi| + 1] \tag{19}$$

con lo que para las permutaciones que se crean de π vía $\pi \star (1)$ serán:

$$(|\pi| + 1)\, \chi(\pi) + \sum_{0 \leq j \leq |\pi|+1} [j = |\pi| + 1] = (|\pi| + 1)\, \chi(\pi) + 1 \tag{20}$$

Esto da para la fución generatriz cumulativa:

$$\begin{align}
M(z) &= \sum_{\pi \in \mathscr{P}} \chi(\pi) \frac{z^{|\pi|}}{|\pi|!} \tag{21} \\
&= \chi(\epsilon) + \sum_{\pi \in \mathscr{P}} \big((|\pi|+1)\,\chi(\pi)+1\big) \frac{z^{|\pi|+1}}{(|\pi|+1)!} \nonumber \\
&= \sum_{\pi \in \mathscr{P}} \chi(\pi) \frac{z^{|\pi|+1}}{|\pi|!} + \sum_{\pi \in \mathscr{P}} \frac{z^{|\pi|+1}}{(|\pi|+1)!} \nonumber \\
&= zM(z) + \sum_{\pi \in \mathscr{P}} \frac{z^{|\pi|+1}}{(|\pi|+1)!} \tag{22}
\end{align}$$

Hay $k!$ permutaciones de tamaño k, sumando sobre tamaños resulta:

$$\begin{align*}
&= zM(z) + \sum_{k \geq 0} k! \frac{z^{k+1}}{(k+1)!} \\
&= zM(z) + \sum_{k \geq 0} \frac{z^{k+1}}{k+1}
\end{align*}$$

Despejando:

$$M(z) = \frac{1}{1-z} \ln \frac{1}{1-z} \tag{23}$$

Reconocemos en (23) la función generatriz de los números harmónicos:

$$\begin{align}
M(z) &= \sum_{n \geq 1} H_n z^n \nonumber \\
H_n &= \sum_{1 \leq k \leq n} \frac{1}{k} \tag{24} \\
&= \ln n + \gamma + O(1/n) \tag{25}
\end{align}$$

donde $\gamma = 0{,}5772156649$ es la constante de Euler.

O sea, asintóticamente el número de asignaciones en el método de selección es mínimo 0, máximo $3n$, promedio $3 \ln n$.