الفصل2

خوارزمياتالفرز

2.1مقدمة

تعدخوارزميات الفرز إحدى الركائز الأساسية لعلوم الكمبيوتر وتلعب دوراً حيوياً في المعالجة الفعالة للبياناتومعالجتها. الفرز، الذي يتضمن إعادة ترتيب العناصر في مجموعة بترتيب معين، هو عملية تستخدمبشكل متكرر في العديد من مجالات الحوسبة، مثل استرجاع المعلومات، وتحليل البيانات، وقواعدالبيانات وغيرها الكثير. في هذا الفصل، سوف نستكشف مجموعة من التقنيات المتطورة والفعالةلتنظيم مجموعات البيانات بطريقة منظمة. سنغطي خوارزميات الفرز الكلاسيكية، مثل فرز التحديد،وفرز الإدراج، وفرز الفقاعات، بالإضافة إلى طرق أكثر تقدماً مثل الفرز السريع، والفرز بالدمج. سوف‌نقوم بدراسة الاختلافات بين الأساليب من حيث الفعالية والتعقيد للسماح للقراء باتخاذ خيارات مستنيرةحول اختيار أفضل طريقة بناءً‌على السياق المحدد للمشكلة المطروحة.

2.2العرض

أيضاًلديهجدولن‌عناصر. الفرز يتكون من إعادة تنظيم عناصرلديهبحيث تكون عناصرلديهيتم طلبها. يمكن أنيكون ترتيب العناصر تصاعدياً أو تنازلياً، أبجدياً أو رقمياً، أو حتى شخصياً، كل هذا يتوقف على سياق المشكلةالمراد حلها وطبيعة البيانات المراد فرزها. على سبيل المثال، في حالة مصفوفة من الأعداد الصحيحة،قد نرغب في فرز العناصر بترتيب تصاعدي، بينما في حالة مصفوفة من السلاسل، قد نرغب في‌فرز العناصر بالترتيب الأبجدي. يمكن لجميع خوارزميات الفرز استيعاب أي نوع بيانات، طالما تم تحديدترتيب العناصر (يجب أن تكون العناصر قابلة للمقارنة).

2.3فرز الفقاعات

يعدفرز الفقاعات أحد أبسط خوارزميات الفرز وأكثرها سهولة. يعتمد على فكرة التكرار عبر المصفوفة عدة مرات،ومقارنة العناصر المتجاورة وتبديلها إذا كانت خارج الترتيب. تتوقف الخوارزمية عند اجتياز الجدول دونإجراء أي تبادلات. يعد فرز الفقاعات خوارزمية فرز مستقرة، مما يعني عدم تغيير ترتيب العناصر المتساوية.يعد فرز الفقاعات خوارزمية فرز موضعية، مما يعني أنها لا تتطلب ذاكرة إضافية لأنها تقوم بإجراءالمقارنات والمبادلات مباشرة في مصفوفة الإدخال.

2.3.1مثال

النظرفي الجدولأ = [5، 1، 4، 2، 8]للفرز بترتيب تصاعدي. تتم عملية فرز الفقاعات على النحو التالي:

— التمريرةالأولى:نحن نجتاز المصفوفة ونقارن العناصر المجاورة. إذا كان العنصر أكبر من العنصر الذييليه، فإننا نستبدله. لاحظ هنا أنه يتم اختيار عملية المقارنة بناءً على ترتيب الفرز المطلوب. إذاأردنا فرز العناصر بترتيب تصاعدي أو حتى فرز نوع آخر من البيانات بترتيب أكثر تعقيداً، فيجب تعديلاختبار الترتيب فقط (عملية المقارنة)، وتبقى بقية الخوارزمية دون تغيير. في مثالنا، نقارن 5 و1، 5أكبر من1,وبالتالي فإن العناصر ليست بالترتيب، لذلك سنقوم بتبديلها. يصبح الجدول:

أ= [1، 5، 4، 2، 8].

ثمنقارن5و4, 5أكبر من4,العناصر ليست مرتبة، نتبادلها فيصبح الجدول:

أ= [1، 4، 5، 2، 8].

نفسالشيء يحدث للعناصر5و2,نستبدلهم ويصبح الجدول:

أ= [1، 4، 2، 5، 8].

للعناصر5و8، 5أصغر من8,العناصر في ترتيبها الصحيح، وبالتالي لا نفعل شيئاً. لاحظ أنه في نهاية الممرالأول، يتم وضع العنصر الأكبر في المصفوفة في نهاية الأخير، وبالتالي، عند كل تمريرة، يمكنناتقليل حجم المصفوفة التي سيتم اجتيازها1.وهذا يعني أنه يمكننا تصفح الأولن -1العناصر بدلامنن للمقطع الثاني، الأولن -2

عناصرالتمريرة الثالثة، وهكذا.

— التمريرةالثانية :نذهب من خلالن -1العناصر الأولى من المصفوفة ومقارنة العناصر المجاورة. إذا كانالعنصر أكبر من العنصر الذي يليه، فإننا نستبدله. في مثالنا، نقارن1و4,ولا يتم تبادل العناصر لأنهامرتبة. ثم نقارن4و2، 4أكبر من2,نستبدلهم ويصبح الجدول:

أ= [1، 2، 4، 5، 8].

ثمنقارن4و5، 4أصغر من5،العناصر مرتبة ويبقى المصفوفة دون تغيير. كما أوضحنا سابقاً، لا نحتاجإلى مقارنة العنصر الأخير في المصفوفة لأنه مرتب بالفعل. وينطبق الشيء نفسه على العنصرقبل الأخير، في المقطع التالي.

— التمريرةالثالثة :نذهب من خلالن -2العناصر الأولى من المصفوفة ومقارنة العناصر المجاورة. وفي هذهالحالة، سوف نلاحظ أن كل شيءن–2 العناصر مرتبة بالفعل، لأنه لم يتم إجراء أي تبادل خلال هذاالمقطع. وبالتالي تتوقف الخوارزمية هنا.

2.3.2التنفيذ

نعرضأدناه تنفيذ خوارزمية فرز الفقاعات في لغة C:

```
1  كثافةالعمليات فقاعة_فرز(كثافة العمليات* لديه، كثافة العمليات ن) {
2      كثافةالعمليات أنا، ي، تمب؛
3      ل(أنا = 0؛ أنا < ن - 1؛ ط ++) {
4          أمر= 1؛
5          ل(ي = 0؛ ي < ن - ط - 1؛ ي++) {
6              لو(أ[ي] > أ[ي + 1]) {
7                  tmp = A]j[;
8                  أ[ي] = أ[ي + 1]؛ أ[ي + 1]
9                  = تمب؛
10                 أمر= 0؛
11             }
12         }
13         لو(أمر) {
14             يعود;
15         }
16     }
17 }
```

الخوارزمية2.1 – فرز الفقاعات – نسخة تكرارية

الإصدارالسابق عبارة عن تطبيق تكراري لخوارزمية فرز الفقاعات. يمكننا أيضاً تنفيذ هذه الخوارزمية بشكلمتكرر، كما يلي:

```
1  كثافةالعمليات(bubble_sort_recكثافة العمليات*لديه،كثافة العملياتن) {
2      كثافةالعملياتأنا، تمة؛
3      لو(ن != 1) {
4          أمر= 1؛
5          ل(أنا = 0؛ أنا < ن - 1؛ ط ++) {
6              لو(أ[i] > أ[i + 1]) {
7                  tmp = A]i[;
8                  أ[i] = أ[i + 1]; tmp = 1[
9                  ;A]i +
10                 أمر=        0;
11             }
12         }
13         لو(!أمر)        {
14 bubble_sort_rec)A, n - 1(;
15         }
16     }
17 }
```

الخوارزمية2.2 - نوع الفقاعة - نسخة متكررة

2.3.3التعقيد

العمليةالأساسية التي تهمنا في خوارزمية فرز الفقاعات هي المقارنة بين عنصرين متجاورين. تحتوي الخوارزميةعلى حلقتين متداخلتين، تضمن الحلقة الأولى أن تقوم الخوارزمية بإجراء عدة تمريرات على المصفوفة(على الأكثرن -1الممرات)، وتضمن الحلقة الثانية تكرار الخوارزمية عبر عناصر المصفوفة. يعتمدعدد العناصر التي يتم اجتيازها في الحلقة الثانية على المسار الحالي، أي أنه في المسار الأول، نجتازجميع عناصر المصفوفة (نقوم بذلكن -1المقارنات)، في المقطع الثاني، نذهب من خلالن -1 العناصرالأولى من المصفوفة، وهكذا. في كل تكرار في الحلقة الثانية، يتم إجراء مقارنة واحدة فقط. تتوقفالخوارزمية عندما تكون جميع العناصر مرتبة أثناء التمريرة الأخيرة. ولكن في أسوأ الحالات (يتم فرزالمصفوفة بترتيب عكسي)، فإننا نقوم بالتنفيذن−1الممرات. وبالتالي فإن عدد المقارنات التي تم إجراؤهافي أسوأ الحالات هو:

$$\text{تفقاعة}(\text{ن}) = (\text{ن} - 1) + (\text{ن} - 2) + \cdots + 1 = \sum_{\text{أنا}=1}^{\text{ن}-1} \text{أنا} = \frac{(\text{ن}-1)\,\text{ن}}{2} = \frac{\text{ن}^2}{2} - \frac{\text{ن}}{2} \in \text{O}(\text{ن}^2) \tag{2.1}$$

وبالتاليفإن الخوارزمية لديها تعقيد من الدرجة الثانية، أي.على2).

يمكنناتطبيق نفس المنطق على النسخة العودية من الخوارزمية، مع ملاحظة أن عدد المقارنات التي تم إجراؤهاالمجموعة من الحجمنشرق ن−1مقارنة. ثم نستدعي الدالة بشكل متكرر بمصفوفة من الحجم

ن−1وهكذا. تتوقف العملية العودية عندما يتم فرز المصفوفة، أو في أسوأ الأحوال، عندما نقوم بإجراء استدعاءمتكرر بمصفوفة ذات حجم1 (وهذا يعني بعدن−1المكالمات العودية).

2.4الفرز حسب الاختيار

الفرزبالتحديد هو خوارزمية فرز تتضمن البحث عن أصغر عنصر في المصفوفة ووضعه في الموضع الأول (في حالة الفرز التصاعدي)، ثم البحث عن ثاني أصغر عنصر ووضعه في الموضع الثاني، وهكذا على. خوارزميةفرز التحديد هي أيضاً خوارزمية موضعية، مما يعني أنها لا تتطلب ذاكرة إضافية لإجراء الفرز. تتوقفالخوارزمية عندما تكون جميع العناصر مرتبة.

2.4.1مثال

سنوضحكيفية عمل خوارزمية الفرز بالاختيار باستخدام المثال التالي:

أ= [5، 1، 4، 2، 8].

— أثناء التمريرة الأولى، نبحث عن أصغر عنصر في المصفوفة، وهو1,ونستبدله بالعنصر الموجود في الموضعالأول من المصفوفة. يصبح الجدول:

أ= [1، 5، 4، 2، 8].

— أثناء التمريرة الثانية، نبحث عن أصغر عنصر في المصفوفة، ولكن هذه المرة نتجاهل العنصر الأول.أصغر عنصر من الموضع الثاني هو2,نستبدله بالعنصر الموجود في الموضع الثاني من المصفوفة.يصبح الجدول:

أ= [1، 2، 4، 5، 8].

لاحظأنه في هذه المرحلة تم فرز المصفوفة بالفعل، ولكن على عكس خوارزمية فرز الفقاعات، ليسلدى خوارزمية فرز التحديد أي طريقة لمعرفة ما إذا تم فرز المصفوفة أم لا، لذلك تستمر في تنفيذالمقاطع التالية، مما يلغي ك−1العناصر الأولى من المصفوفة في كل مسار، حيثكهو رقم المقطعالحالي. سيبقى الجدول دون تغيير حتى التمريرة الأخيرة.

2.4.2التنفيذ

نعرضأدناه تنفيذ الإصدار التكراري لخوارزمية فرز التحديد بلغة C:

```
1  كثافةالعملياتاختيار_فرز(كثافة العمليات*لديه،كثافة العملياتن) {
2      كثافةالعملياتأنا، ي، ind_min، تمب؛ لأنا(
3          = 0؛ أنا < ن - 1؛ ط ++) {
4              ind_min = i;
5          لأ(ي = ط + 1؛ ي < ن؛ ي++) {
6              لو(أ[ي] < أ[ind_min]) {
7                  ind_min = j;
8              }
9          }
10         لو( ط =! ind_min) {
11             تمب= أ[أنا]؛
12             أ[i] = أ[ind_min];
13             أ[ind_min] = تمة؛
14         }
15     }
16 }
```

الخوارزمية2.3 – فرز التحديد – النسخة التكرارية

نعرضأدناه تنفيذ الإصدار العودي لخوارزمية فرز التحديد بلغة C:

```
1  كثافةالعملياتSelect_sort_recكثافة العمليات*لديه،كثافة العملياتن) {
2      كثافةالعملياتأنا، ind_min،
3          تمب؛ لو(ن != 1) {
4              ind_min = 0;
5          لأ(أنا = 1؛ أنا <ن؛ ط ++) {
6              لو(أ[i] < أ[ ind_min]) {
7                  ind_min = i;
8              }
9          }
10         لو( 0 =! ind_min) {
11             تمب= أ[0];
12             أ[0] = أ[ind_min];
13             أ[ind_min] = تمة؛
14         }
15         Selection_sort_rec)A + 1, n - 1(;
16     }
17 }
```

الخوارزمية2.4 - الترتيب حسب الاختيار - نسخة متكررة

2.4.3التعقيد

العمليةالأساسية التي نهتم بها هنا هي أيضاً المقارنة. تحتوي الخوارزمية على حلقتين متداخلتين، تضمن الحلقةالأولى أداء الخوارزميةن -1يمر فوق المصفوفة، وتتكرر الحلقة الثانية عبر عناصر المصفوفة بدءاً منالموضعأنا+1 (أوأناهو رقم المقطع الحالي) للعثور على الحد الأدنى. يعتمد عدد المقارنات التي يتم إجراؤهافي الحلقة الثانية على المقطع الحالي، أي أننا في المقطع الأول نمرن -1العناصر، في الممر الثاني،نمرن -2العناصر، وما إلى ذلك. وبالتالي فإن عدد المقارنات التي تم إجراؤها هو:

$$ت_{اختيار}(ن) = (ن -1) + (ن -2) + \cdots + 1 = \sum_{أنا=1}^{ن-1} أنا = \frac{ن(ن -1)}{2} = \frac{ن^2}{2} - \frac{ن}{2} \quad (2.2)$$

وبالتاليفإن الخوارزمية لديها تعقيد من الدرجة الثانية، أي.على2).

يمكنناتطبيق نفس المنطق على النسخة العودية من الخوارزمية بطريقة مشابهة لخوارزمية فرز الفقاعات.نصل دائماً إلى نفس التعقيد التربيعي.

2.4.4المقارنة مع نوع الفقاعة

عندمانعتبر المقارنة هي العملية الأساسية، فإن خوارزمية فرز التحديد تنفذ دائماً نفس عدد المقارنات بغضالنظر عن ترتيب العناصر في المصفوفة. بينما قد تقوم خوارزمية فرز الفقاعات بإجراء مقارنات أقل إذاوجدت أن المصفوفة قد تم فرزها في نهاية التمريرة. ومع ذلك، يمكن أن تكون خوارزمية فرز التحديد أكثركفاءة من خوارزمية فرز الفقاعات في بعض الحالات، لأنها تؤدي عمليات تبادل أقل من خوارزمية فرزالفقاعات.

تغييرالمنظور

عندمانعتبر التبادل بمثابة العملية الأساسية، فإن خوارزمية فرز الاختيار هي تبادل واحد فقط لكل تمريرة علىالأكثر. وبما أن عدد المقاطع هون -1,تعقيد خوارزمية فرز الاختيار في هذا السياقص (ن) (التعقيد الخطي). في حين أن خوارزمية فرز الفقاعات في أسوأ الأحوال تقوم بمبادلة واحدة لكل مقارنة (في أسوأ الحالات- يتم فرز المصفوفة بشكل عكسي)، وعدد المقارنات هو نفس عدد المقايضات، فإن تعقيد خوارزميةفرز الفقاعات في هذا السياق هوعلى2) (التعقيد التربيعي). يشرح هذا السلوك الفرق في الأداء بينالخوارزميتين.

-تكون خوارزمية فرز التحديد أكثر كفاءة من خوارزمية فرز الفقاعات إذا تم تشغيلها على بنية أجهزة حيثيكون المبادلة أكثر تكلفة من المبادلة.

المقارنة،خاصة على المعالجات ذات السجلات الصغيرة أو ذاكرة التخزين المؤقت الصغيرة.

-تعد خوارزمية فرز الفقاعات أكثر كفاءة من خوارزمية فرز التحديد إذا تم تشغيلها على بنية أجهزة حيث تكونالمقارنة أكثر تكلفة من المبادلة. ستؤدي عملية المقارنة عند استخدامها في حالة ما إلى قفزة مشروطة،وهي عملية مكلفة في وقت التنفيذ، خاصة على المعالجات ذات خطوط الأنابيب العميقةأو متنبئ فرعي غير فعال.

يمكنأن يعتمد الفرق بين الخوارزميتين أيضاً على طبيعة البيانات المراد فرزها. إذا كانت البيانات مصنفةبالفعل (تقريباً)، فستكون خوارزمية فرز الفقاعات أكثر كفاءة من خوارزمية فرز التحديد لأنها تقوم بعددأقل من التمريرات عبر المصفوفة. وبالتالي فإن الخوارزمية المستخدمة بين الاثنين تعتمد على طبيعةالبيانات التي سيتم فرزها وبنية الأجهزة التي سيتم تنفيذ الخوارزمية عليها.

2.5نوع الإدراج

خوارزميةفرز الإدراج هي خوارزمية فرز تتضمن إدراج كل عنصر من عناصر المصفوفة في مكانه المناسب فيمصفوفة مفروزة. تبدأ الخوارزمية باعتبار أن العنصر الأول من المصفوفة مفروز، ثم تقوم بإدخال العنصرالثاني مكانه في المصفوفة المصنفة، ثم تقوم بإدخال العنصر الثالث مكانه في المصفوفة المصنفة،وهكذا حتى تنتهي جميع العناصر يتم إدراج عناصر المصفوفة في مكانها. خوارزمية فرز الإدراج هيخوارزمية مستقرة (يتم الحفاظ على ترتيب العناصر المتساوية) وفي مكانها (لا تتطلب الخوارزمية ذاكرةإضافية). تعتمد هذه الخوارزمية على تعقيد الإدراج في مصفوفة مرتبة، وهي تحت الخط (لوغاريتمي) لتحسين تعقيد خوارزمية الفرز.

2.5.1مثال

نعرضأدناه مثالاً لتشغيل خوارزمية فرز الإدراج على المصفوفة المكونة من 5 عناصر التالية:

أ= [{5}، 2، 4، 6، 1]

الطاولةلديهوينقسم إلى قسمين، الجزء الأول هو المصفوفة التي تم فرزها، والجزء الثاني هو بقية المصفوفة.حجم المصفوفة التي تم فرزها هو في البداية 1، وحجم بقية المصفوفة هو في البدايةن -1 (أو نهو حجم المصفوفة). تقوم الخوارزمية أولاً بإدراج العنصر الثاني من المصفوفة في المصفوفة التي تم فرزها،ثم تقوم بإدراج العنصر الثالث من المصفوفة في المصفوفة التي تم فرزها، وهكذا حتى يتم إدراج جميعالعناصر

يتمإدراج الجدول في مكانهم. نلاحظ الجزء الذي تم فرزه من المصفوفة من خلال وضعه بين قوسين متعرجين.

— المقطع1:في المسار الأول، المصفوفة التي تم فرزها هي [{5 }]وبقية الجدول [2، 4، 6، 1].
الخطوةالأولى هي إدراج العنصر الثاني من المصفوفة (القيمة2)في المصفوفة المرتبة نحصل على النتيجة:

أ= [{2، 5}، 4، 6، 1]

.

— المقطع2:في المسار الثاني، المصفوفة التي تم فرزها هي [{2، 5}]وبقية الجدول هو [4، 6، 1].
سوفنقوم بإدراج القيمة4في المصفوفة المرتبة نحصل على النتيجة:

أ= [{2، 4، 5}، 6، 1]

.

— المقطع3:وفي الممر الثالث، سوف نقوم بإدراج القيمة6في المصفوفة المرتبة نحصل على النتيجة:

أ= [{2، 4، 5، 6}، 1]

.

— المقطع4:يتكون المسار الرابع من إدراج العنصر الأخير في المصفوفة (القيمة1)في الجزء المرتب منالمصفوفة، مما يعطينا المصفوفة المرتبة التالية:

أ= [{1، 2، 4، 5، 6}]

.

2.5.2التنفيذ

قبلتنفيذ خوارزمية فرز الإدراج، من المهم جداً تنفيذ وفهم عملية الإدراج في مصفوفة مفروزة. الإصدار الساذجللإدراج في مصفوفة مفروزة هو التكرار عبر المصفوفة بحثاً عن موضع الإدراج (العنصر الأول الأكبرمن العنصر المراد إدراجه في حالة الفرز التصاعدي)، ثم إزاحة جميع عناصر المصفوفة من موضع الإدراجمربع واحد إلى اليمين، ثم أدخل العنصر في مكانه. يتم تنفيذ هذا الإصدار من الإدراج بلغة C أدناه:

الإدراجفي مصفوفة مرتبة

```
1 فارغإدراج (كثافة العملياتمصفوفة,كثافة العملياتمقاس،كثافة العملياتعنصر) {
2 كثافةالعملياتأنا = 0؛
3 بينما(;i++ )element > ]i[array && size > i ل(كثافة العمليات
4 = الحجم؛ ي > أنا؛ ي--) {
5 المصفوفة[j] = المصفوفة[1 - j];
6 }
7 المصفوفة[i] = العنصر;
8 }
```

الخوارزمية2.5 – الإدراج في مصفوفة مرتبة – التعقيد الخطي

لاحظأن الدالةإدراجيأخذ مجموعة من الأعداد الصحيحة كمعلمةمصفوفة,حجم الطاولةمقاسوالعنصر المرادإدراجهعنصر.الوظيفةإدراجإرجاع المصفوفةصفيف مع العنصرعنصرإدراجها في مكانها على افتراض أنالجدولصفيفمتبوعاً بمربع فارغ يمكن استخدامه للتأكد من أن إزاحة عناصر المصفوفة لا تتجاوز حجمها.

منالممكن تنفيذ إصدار أكثر كفاءة للإدراج في مصفوفة مرتبة باستخدام البحث الثنائي للعثور على موضعإدراج العنصر. البحث الثنائي هو البحث الذي يتكون من تقسيم الجدول إلى قسمين، ثم البحث فيأي جزء يقع العنصر المطلوب البحث عنه، ثم يتم تقسيم الجدول إلى قسمين في كل مرة على النحو التالي:

— إذا كان العنصر المراد إدراجه أكبر من العنصر الموجود في منتصف المصفوفة، فيجب إدراج العنصرفي الجزء الثاني من المصفوفة.

-إذا كان العنصر المراد إدراجه أصغر من العنصر الموجود في منتصف المصفوفة، فيجب إدراج العنصرفي الجزء الأول من المصفوفة.

يتمتكرار نفس العملية حتى يتم العثور على موضع إدراج العنصر، أو ينتهي بنا الأمر بمصفوفة بحجم 0.

يتمتنفيذ إصدار الإدراج في مصفوفة مرتبة باستخدام البحث الثنائي بلغة C أدناه:

```
1 فارغإدراج (كثافة العملياتمصفوفة,كثافة العملياتمقاس،كثافة العملياتعنصر) {
2 كثافةالعملياتط = 0، ي = الحجم - 1، ك؛
3 بينما(أنا <= ي) {
4 ك= (ط + ي) / 2؛
5 لو(المصفوفة [ك] < العنصر) i = k + 1؛ آخريي = ك - 1؛
6
7 }
8 ل(ك = الحجم - 1؛ ك > ط؛ ك--) {
9 المصفوفة[ك] = المصفوفة[ك - 1];
```

```
10      }
11      المصفوفة[i] = العنصر;
12  {
```

الخوارزمية2.6 – الإدراج في مصفوفة مرتبة – التعقيد اللوغاريتمي

يمكنأيضاً تنفيذ عملية البحث عن موضع الإدراج بشكل متكرر، مما يؤدي إلى الإصدار التالي:

```
1  كثافةالعمليات يبحث(كثافة العمليات* مصفوفة,كثافة العمليات أنا،كثافة العمليات ي،كثافة العمليات عنصر) {
2      لو(أنا > ي)يعودأنا؛ كثافة
3      العملياتك = (ط + ي) / 2؛
4      لو(صفيف [ك] <عنصر)يعود;(search)array, k + 1, j, element عودة أخرى;(search)array, i, k - 1, element
5  
6  {
7  
8  فارغإدراج (كثافة العمليات* مصفوفة,كثافة العمليات مقاس،كثافة العمليات عنصر) {
9      كثافةالعمليات;(search)array, 0, size - 1, element i = ل(كثافة
10     العملياتك = الحجم - 1؛ ك > ط؛ ك--) {
11         المصفوفة[ك] = المصفوفة[ك - 1];
12     }
13     المصفوفة[i] = العنصر;
14 {
```

الخوارزمية2.7 – الإدراج في مصفوفة مرتبة – نسخة متكررة

بمجردالانتهاء من تنفيذ وظيفة الإدراج في مصفوفة مرتبة، يمكننا الآن الانتقال إلى فرز الإدراج. سنقوم بالتكرارخلال المصفوفة التي سيتم فرزها عن طريق إدراج كل عنصر في المصفوفة التي تم فرزها، مما يمنحناالتنفيذ التالي:

```
1  فارغالإدراج_الفرز (كثافة العمليات* مصفوفة,كثافة العمليات مقاس) {
2      ل(كثافة العملياتأنا = 1؛ أنا <الحجم؛ ط++)
3          إدراج(صفيف، أنا، مجموعة [i])؛
4  {
```

الخوارزمية2.8 - فرز الإدراج

2.5.3التعقيد

نحننعتبر المقارنة بمثابة العملية الأساسية لخوارزمية فرز الإدراج. يعتمد تعقيدها بشكل أساسي على مدىتعقيد وظيفة الإدراج في مصفوفة مرتبة. في الحالة التي نستخدم فيها البحث الثنائي للعثور على موضعإدراج العنصر، يكون تعقيد دالة الإدراج لوغاريتمياً، نظراً لأن البحث الثنائي يقسم المصفوفة إلى جزأينفي كل مرة. بمعنى آخر، عدد المقارنات التي تم إجراؤها للعثور على

إدراجالموقف في مجموعة مرتبة من الحجمنيقتصر على عدد المرات التي يمكننا القسمة فيهانبمقدار 2،وهو ما يعطي التعقيد اللوغاريتمي في أسوأ الحالاتيا(سجل2ن).

يتمتنفيذ الإدراجنمرات في حالة نوع الإدراج، مما يعطي تعقيداًعلىسجلن)في أسوأ الأحوال. هذا التعقيدمتشائم بعض الشيء، لأن حجم المصفوفة التي يتم إدراج العناصر فيها يبدأ من 1 ويزداد في كل مرة،مما يعني أن عدد المقارنات التي يتم إجراؤها لإدراج العناصر يتم الحصول عليها من خلال الصيغة التالية:

ت الإضافيةقأورت(ن) =سجل2 1 + سجل2 2 + سجل2 3 +· · · +سجل2ن (2.3)

لكننتيجة هذا المجموع تكون دائماً أقل من المجموع التالي:

سجل2ن+سجل2ن+سجل2ن+· · · +سجل2ن = نسجل2ن (2.4)

ومنهنا التعقيدعلىسجل2ن).

2.5.4المقارنة مع خوارزميات الفرز الأخرى

تعتبرالمقارنة بين خوارزميات الفرز مهمة صعبة، لأنه يمكن اعتبار العديد من العمليات عمليات أساسية. فيكثير من الأحيان نعتبر عدد المقارنات بمثابة العملية الأساسية، وفي هذه الحالة يبدو أن فرز الإدراج أكثركفاءة من فرز التحديد وفرز الفقاعات. ومع ذلك، فإن الفرز بالإدراج ينفذ عمليات ذاكرة أكثر من الفرز بالتحديد.في الواقع، في أسوأ الحالات، عندما تعتبر عمليات الذاكرة عمليات أساسية، يكون فرز الإدراج معقداًعلى2) ، وهو نفس التعقيد مثل الفرز الفقاعي، في حين أن الفرز بالتحديد له تعقيدعلى)،وهو أكثر كفاءةمن الخوارزميتين الأخريين. بالإضافة إلى ذلك، فإن خوارزمية فرز الإدراج قابلة للتكيف، أي إذا تم فرزالمصفوفة تقريباً، فسيكون عدد العمليات المنفذة صغيراً جداً.

فيالختام، فإن الخوارزمية الأكثر فعالية من بين الثلاثة التي شوهدت حتى الآن تعتمد على الموقف، والأسئلةالتي يجب طرحها هي:

-هل تم فرز المصفوفة جزئياً؟

-هل عملية الذاكرة أغلى من عملية المقارنة؟

-هل متنبئ الفرع فعال؟

-هل عدد المسجلين محدود؟

-هل الذاكرة المؤقتة كافية؟

2.6دمج الفرز

دمجالفرز هو خوارزمية متكررة تقسم المصفوفة التي سيتم فرزها إلى جزأين، وتفرز كل جزء، ثم تدمجها للحصولعلى المصفوفة التي تم فرزها. تأخذ وظيفة الدمج صفيفين مفروزين وتدمجهما في صفيف واحدمفروز دائماً. خوارزمية الفرز المدمج هي خوارزمية فرز خارجية، مما يعني أنها يجب أن تستخدم مساحةالذاكرة الخارجية لتخزين البيانات المراد فرزها. تتوقف الاستدعاءات العودية عندما تحتوي المصفوفةالمراد فرزها على عنصر واحد فقط، لأن المصفوفة المكونة من عنصر واحد يتم فرزها دائماً.

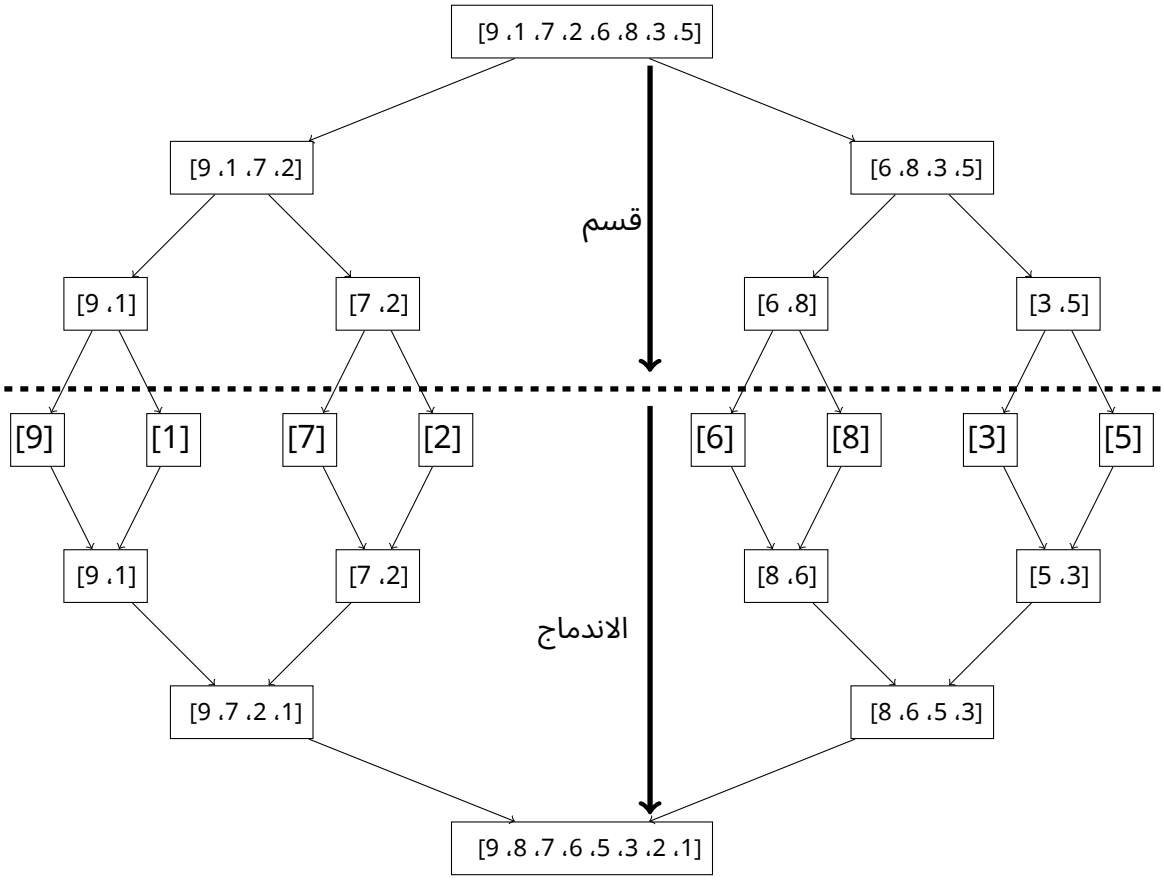
يعدفرز الدمج خوارزمية فرز متقدمة إلى حد ما وهي فعالة جداً عندما يتعلق الأمر بفرز مصفوفة كبيرة. كماأنها فعالة جداً عندما يتعلق الأمر بفرز البيانات المخزنة على وسيط خارجي، لأنها تقوم بنسخ جزء من البياناتليتم فرزها في الذاكرة الرئيسية في كل مرة، مما يجعل من الممكن تجنب الوصول المتكرر إلى الذكرياتالخارجية التي تكون بطيئة جداً .

2.6.1مثال

سنوضحكيفية عمل خوارزمية الفرز المدمج باستخدام مثال. سنقوم بفرز الجدولأ = [5، 3، 8، 6، 2، 7، 9،1].يتم توضيح عملية فرز الدمج في الشكل2.1. لقد اخترنا على وجه التحديد مخطط الحجم2كلتسهيل الشرح.ومع ذلك، تعمل خوارزمية الفرز المدمج أيضاً مع المصفوفات ذات الحجم وليس مع قوى 2. في الجزءالعلوي من الشكل2.1، يتم تقسيم المصفوفة المراد فرزها إلى جزأين متساويين في كل مرة حتى نحصلعلى مصفوفات بالحجم 1 (والتي يتم فرزها حسب التعريف). في الجزء السفلي، يتم دمج المصفوفاتالتي تم فرزها حتى يتم الحصول على المصفوفة التي تم فرزها بشكل نهائي.

2.6.2التنفيذ

يتمتنفيذ خوارزمية فرز الدمج باستخدام دالة متكررة تأخذ المصفوفة التي سيتم فرزها بحجمها كمعلمة. تقومالدالة العودية بتقسيم المصفوفة إلى جزأين، ثم تستدعي بشكل متكرر دالة الفرز في كل جزء. عندما يكونحجم المصفوفة يساوي 1، تقوم الدالة بإرجاع المصفوفة. عند إرجاع كلا المصفوفتين المفرزتين، يتماستدعاء دالة أخرى لدمجهما في مصفوفة واحدة لا تزال مفروزة. سنبدأ أولاً بتنفيذ وظيفة الدمج التي تأخذصفيفتين فرعيتين مفروزتين متتاليتين كمعلمات وتدمجهما في مصفوفة مفروزة واحدة تشغل نفسالمساحة التي تشغلها المصفوفتان المدمجتان. يجب أن تستخدم الوظيفة مساحة الذاكرة الخارجية لتخزينالمصفوفة المدمجة ثم نسخها إلى المصفوفة



الشكل2.1 -دمج الفرز

أولي.يظهر تنفيذ وظيفة الدمج أدناه.

```
1  فارغ دمج(كثافة العمليات* أولاً، كثافة العمليات* ثانية، كثافة العمليات الحجم_أولاً، كثافة العمليات الحجم_الثانية) {
2      كثافةالعمليات* تم الدمج = مالوك ((الحجم_الأول + الحجم_الثاني) *sizeof(كثافة العمليات)); كثافة
3  العمليات ط = 0، ي = 0، ك = 0؛
4      بينما(أنا <حجم_الأول && ي <حجم_الثانية) {
5          لو(الأول[i] <الثاني[ي]) {
6              اندمجت[ك] = أولا[أنا]؛
7              أنا++;
8          } آخر {
9              اندمجت[ك] = الثانية[ي]؛
10             ي++;
11         }
12         ك++;
13     }
14     بينما(أنا<size_first) {
15         Merged]k[ = first]i[;
16         أنا++;
17         ك++;
18     }
```

```
19      بينما(ي <حجم_الثانية) {
20          Merged[k] = Second[j];
21          ي++;
22          ك++;
23      }
24      ل(أنا = 0؛ أنا <حجم_الأول + حجم_الثانية؛ ط ++) {
25          first[i] = merged[i];
26      }
27      مجاني(مدمج);
28  }
```

الخوارزمية2.9 – دمج صفيفين مفروزين

الآنسوف نقوم بتنفيذ وظيفة فرز الدمج. تأخذ هذه الوظيفة العودية المصفوفة المراد فرزها وحجمها كمعلمات.إذا كان حجم المصفوفة يساوي 1، فإن الدالة ترجع نفس المصفوفة. بخلاف ذلك، تقوم الدالة بتقسيمالمصفوفة إلى جزأين ثم تستدعي دالة الفرز بشكل متكرر في كل جزء. عندما يتم إرجاع كلا المصفوفتينالمصنفتين، نستخدم وظيفة الدمج في الخوارزمية2.9لدمج دمجهم. يظهر تنفيذ وظيفة فرز الدمجأدناه:

```
1   فارغ دمج_الفرز(كثافة العمليات* مصفوفة, كثافة العمليات مقاس) {
2       لو(الحجم ! = 1) {
3           كثافةالعمليات الحجم / 2؛ = size_first
4           كثافةالعمليات؛size_first - الحجم = size_sec كثافة
5           العمليات*الأول = المصفوفة؛
6           كثافةالعمليات*،؛ دمج_الفرز (الأولsize_first +
7                   size_first(;        Second = array
8           دمج_فرز (الثانية،           ؛size_sec)
9           دمج(الأول، الثاني،          ؛size_first، size_first
10      }
11  }
```

الخوارزمية2.10 – دمج الفرز

2.6.3التعقيد

أمابالنسبة للخوارزميات الأخرى فنعتبر أن عملية المقارنة هي العملية الأكثر تكلفة. لذلك، سنقوم بتقدير مدىتعقيد خوارزمية الفرز المدمج عن طريق حساب عدد المقارنات التي تم إجراؤها. يعتمد تعقيد خوارزميةفرز الدمج على (1) مدى تعقيد وظيفة الدمج و (2) عدد المكالمات العودية، أي عمق العودية. تعقيدوظيفة الدمج خطي، لأنها تمر عبر كلا المصفوفتين ليتم دمجها مرة واحدة فقط. لا يمكن أن يتجاوز عددالاستدعاءات العودية عدد المرات التي يمكننا فيها تقسيم حجم المصفوفة على 2 حتى نحصل على 1،وهو ما يساويسجل2نأونهو حجم المصفوفة الأولية. لأنه في كل مستوى من مستويات العودية، تتم مقارنةكل عنصر بعنصر واحد

مرات على الأكثر(ن مقارنات على كل مستوى)، وأن عمق العودية هو سجل2ن، تعقيد خوارزمية فرز الدمج هو في حدود على·سجل2ن).

إذا كنا مهتمين بعمليات الذاكرة، فإن خوارزمية الفرز المدمج دائماً ما تكون معقدة من حيث الترتيب على· سجل2ن)، لأنه في كل مستوى من مستويات العودية، ن يتم نسخ العناصر إلى مصفوفة خارجية، ثم يتم نسخها مرة أخرى إلى المصفوفة الأولية (ت(ن) =2ن). عمق العودية هو سجل2ن، وبالتالي فإن تعقيد خوارزمية الفرز المدمج هو في حدود على·سجل2ن).

2.6.4 المقارنة مع الخوارزميات الأخرى

تعد خوارزمية الفرز المدمج أكثر كفاءة من خوارزميات الفرز الأخرى التي رأيناها حتى الآن. في الواقع، فإن تعقيد خوارزمية الفرز المدمج هو في حدود على·سجل2ن)، مهما كانت العملية تعتبر. هذا هو التعقيد الأمثل لخوارزميات فرز المقارنة. وبالتالي فإن خوارزمية فرز الدمج تكون أكثر كفاءة من خوارزميات فرز الفقاعات والإدراج والاختيار، خاصة عندما يكون حجم المصفوفة المراد فرزها كبيراً.

2.7 الفرز السريع (الفرز السريع)

عبارة عن خوارزمية فرز مقارنة تستخدم تقنية Quicksort فرق تسد. مثل فرز الدمج، فهو يعتمد على تقنية التقسيم والتي تتمثل في تقسيم الجدول إلى قسمين، ثم فرز كل جزء على حدة. خوارزمية الفرز السريع هي خوارزمية متكررة تستخدم وظيفة التقسيم لتقسيم المصفوفة إلى قسمين. تختار وظيفة التقسيم عنصراً من المصفوفة تسمى محور، ثم يضع جميع العناصر الأصغر من المحور على يساره وجميع العناصر الأكبر من المحور على يمينه. خوارزمية الفرز السريع هي أيضاً خوارزمية موضعية، لأنها لا تتطلب مصفوفة خارجية لإجراء الفرز.

2.7.1 مثال

سنوضح كيفية عمل خوارزمية الفرز السريع في الجدول التالي:

أ= [5، 3، 7، 2، 1، 4، 6]

أولاً سنختار المحور. يمكن اختيار الأخير بطرق مختلفة، ولكن الخيار الأكثر شيوعاً هو اختيار العنصر الأخير من المصفوفة. في لدينا

علىسبيل المثال، المحور يساوي6.سنقوم بعد ذلك باجتياز المصفوفة من اليسار إلى اليمين، ووضع جميعالعناصر الأصغر من المحور على يسارها. وعندما نجد عنصراً أصغر من المحور، نستبدله بالعنصر الأولالأكبر من المحور. عندما ننتهي من الجدول بأكمله، نستبدل المحور بالعنصر الأول الأكبر منه. فيصبحالجدول بالتالي:

أ= [{5، 3، 2، 1، 4}،6،{ 7 }]

بعدذلك، نطبق نفس الإجراء على جزأين من الجدول مفصولين بالمحور. تم فرز الجانب الأيمن من المصفوفةبالفعل لأنه يحتوي على عنصر واحد فقط. ولذلك فإننا نطبق الإجراء على الجزء الأيسر من الجدول.يتم اختيار العنصر الأخير من المصفوفة الفرعية كعنصر محوري، وقيمته هي4,وبعد هذه الخطوة يصبحالجدول كما يلي:

أ= [{3، 2، 1}،4، {5}، 6، 7]

نحننطبق نفس الإجراء على المصفوفة الفرعية من [3، 2، 1]،يتم اختيار العنصر الأخير كمحور، وقيمته هي1,وبعد هذه الخطوة يصبح الجدول كما يلي:

أ= [1، {2، 3}، 4، 5، 6، 7]

فيالخطوة الأخيرة، نطبق نفس الإجراء على المصفوفة الفرعية [2، 3]،لقد تم فرزه بالفعل، ولكن لإنهاء المثال،سنظل نختار العنصر الأخير كمحور، وقيمته هي3،وبعد هذه الخطوة يصبح الجدول كما يلي:

أ= [1، {2}،3،4، 5، 6، 7]

فيالنهاية هذه الخطوة الأخيرة، يتم فرز الجدول.

2.7.2التنفيذ

فيمايلي تنفيذ خوارزمية الفرز السريع:

```
1  فارغفرز_سريع(كثافة العمليات*مصفوفة,كثافة العملياتمقاس) {
2      لو(الحجم > 1) {
3          كثافةالعملياتالمحور = المصفوفة [الحجم - 1]؛ كثافة
4          العملياتأنا = 0؛
5          كثافةالعملياتي = الحجم
6          -2؛ بينما(أنا <= ي) {
7              لو(المصفوفة[i]> المحورية && المصفوفة[j] < المحورية) {
8                  كثافةالعمليات;]i[tmp = array
```

```
9           المصفوفة[i]   = صفيف[ي];
10          صفيف[ي]   = تمة؛
11          أنا++;
12          ي--؛
13      }وإلا إذا(المصفوفة [i] <= المحور) {
14          أنا++;
15      }وإلا إذا(المصفوفة [ي] >= المحور) {
16          ي--؛
17      }
18  }
19  كثافةالعمليات
20  ;(Quick_sort)array, i المصفوفة[ = tmp; Quick_sort)array, i(;
21  الحجم- 1] ;array]size - 1[ = 1
22  tmp = array]i[; array]i[ =
23  Quick_sort)array + i + 1, size - i - 1(;
24  }
25  {
```

2.7.3التعقيد

دعونانعتبر عملية المقارنة هي العملية الأساسية. يكون تعقيد خوارزمية الفرز السريع في أسوأ الحالات (عندمايتم فرز المصفوفة بالفعل) في حدود على2)، لأنه في كل مستوى من مستويات التكرار، تتم مقارنة كلعنصر بالمحور مرة واحدة على الأكثر (ن - طمقارنات مثلأناهو مستوى العودية). عمق العودية هون، وبالتاليفإن تعقيد خوارزمية الفرز السريع هو في حدود على2). يكون تعقيد خوارزمية الفرز السريع في أفضلالأحوال (عندما يكون المحور دائماً في منتصف المصفوفة) في حدودأوم(ن·سجل2ن)،لأنه في كل مستوىمن مستويات العودية، تتم مقارنة كل عنصر بالمحور مرة واحدة على الأكثر (ن - طمقارنات مثل أناهو مستوى العودية). عمق العودية هوسجل2ن،وبالتالي فإن أفضل تعقيد لحالة خوارزمية الفرز السريع هومن ترتيبأوم(ن·سجل2ن).يبلغ تعقيد خوارزمية الفرز السريع في المتوسط حواليّن·سجل2ن،لأنه علىصفائف عشوائية من الحجمن،فمن غير المرجح أن يكون هناك مصفوفة تم فرزها بالفعل.

2.7.4المقارنة مع فرز الدمج

علىالرغم من أن التعقيد الأسوأ لخوارزمية الفرز السريع هو في حدودعلى2)في أسوأ الحالات، تكون هذه الخوارزميةفي المتوسط أكثر كفاءة من خوارزمية الفرز المدمج. في الواقع، متوسط التعقيد لخوارزمية الفرزالسريع هو في حدود على·سجل2ن)،نفس خوارزمية فرز الدمج. بالإضافة إلى ذلك، فإن خوارزمية الفرزالسريع هي خوارزمية موضعية، بينما تتطلب خوارزمية الفرز المدمج مصفوفة خارجية لإجراء الفرز. بالإضافةإلى ذلك، في أسوأ الحالات بالنسبة لخوارزمية الفرز السريع، لا يتم إجراء أي عملية نسخ، وتتم المقارنةمع المحور

دائماًما تكون النتيجة سلبية، مما يسمح لمتنبئ الفرع بتكييف أداء الخوارزمية وتحسينه.

2.8الاستنتاج

فيهذا الفصل، قدمنا العديد من خوارزميات الفرز. لقد بدأنا بأبسط الخوارزميات لفهمها وتنفيذها، مثلفرز الفقاعات وفرز التحديد. تحتوي هاتان الخوارزميتان على تعقيد بترتيبعلى2)، وبالتالي فهي غير فعالةعلى الطاولات الكبيرة. ثم قدمنا نوع الإدراج، الذي يتميز بترتيب معقدعلى·سجل2ن)عندما يتعلق الأمربالمقارنات، ولكن من أجلعلى2)إذا نظرنا إلى عمليات الذاكرة. ثم قدمنا نوع الدمج، الذي يتميز بدرجةتعقيد الترتيبعلى·سجل2ن)،وهو أسرع بكثير من الخوارزميات السابقة على المصفوفات الكبيرة. أخيراً،تم تقديم خوارزمية الفرز السريع، وهي معقدة في أسوأ الأحوالعلى2)، ولكنه من الناحية العملية أسرعمن فرز الدمج، لأنه موجود في مكانه وأن متوسط التعقيد هو في حدودن·سجل2ن.

يعتمداختيار خوارزمية الفرز التي سيتم استخدامها على عدة عوامل، مثل حجم المصفوفة التي سيتم فرزها،وتعقيد الخوارزمية، وطبيعة البيانات التي سيتم فرزها، وحجم الذاكرة المتوفرة، وما إلى ذلك. من الناحيةالعملية، يعد الفرز السريع هو خوارزمية الفرز الأكثر استخداماً على نطاق واسع، بما في ذلك المكتباتالقياسية للغات البرمجة. ومع ذلك، في بعض الحالات قد تكون خوارزميات الفرز الأخرى أكثر كفاءة.