6- compare between Array and Linked List

**المقارنة بين المصفوفة (Array) والقائمة المترابطة (Linked List):**

**1. الهيكل والتخزين:**

* **المصفوفة (Array):** هي هيكل بيانات ثابت الحجم يتم تخزين العناصر في مواقع متتالية في الذاكرة. يتم الوصول إلى العناصر باستخدام فهرس ثابت يبدأ عادةً من الصفر.
  + **الخصائص:**
    - حجم ثابت بعد التحديد (لا يمكن تغييره بعد الإنشاء).
    - الوصول العشوائي للعناصر بسرعة (O(1)).
    - جميع العناصر مخزنة في مواقع متجاورة في الذاكرة.
* **القائمة المترابطة (Linked List):** هي هيكل بيانات ديناميكي يتكون من مجموعة من العقد (Nodes)، حيث تحتوي كل عقدة على عنصر والمرجع إلى العقدة التالية.
  + **الخصائص:**
    - حجم ديناميكي (يمكن تغيير الحجم أثناء التشغيل).
    - لا يتم تخزين العناصر في مواقع متجاورة.
    - الوصول إلى العناصر بترتيب متسلسل (O(n)).

**2. الأداء:**

* **المصفوفة (Array):**
  + **الولوج العشوائي:** الوصول إلى أي عنصر يتم في وقت ثابت O(1).
  + **إضافة العناصر:** إضافة عنصر جديد يتطلب عادةً إعادة تخصيص المصفوفة إذا كانت ممتلئة (O(n) في أسوأ الحالات).
  + **حذف العناصر:** يتطلب تحريك العناصر المتبقية بعد الحذف (O(n)).
* **القائمة المترابطة (Linked List):**
  + **الولوج العشوائي:** الوصول إلى عنصر معين يستغرق وقتًا خطيًا O(n) حيث نحتاج للتنقل عبر العقد للوصول إلى العنصر المطلوب.
  + **إضافة العناصر:** إضافة عنصر في بداية أو نهاية القائمة يتم بسرعة (O(1)).
  + **حذف العناصر:** حذف عنصر يمكن أن يتم بسرعة أيضًا (O(1)) إذا كان لدينا مرجع إلى العنصر المطلوب.

**3. استخدام الذاكرة:**

* **المصفوفة (Array):**
  + **استهلاك الذاكرة ثابت**: بما أن المصفوفة تحتوي على مساحة ثابتة من الذاكرة، فإن استخدامها للذاكرة يكون ثابتًا.
  + **العيب:** في حالة عدم معرفة حجم البيانات مسبقًا، قد يؤدي ذلك إلى إهدار الذاكرة أو تجاوز الحجم المحدد.
* **القائمة المترابطة (Linked List):**
  + **استهلاك الذاكرة ديناميكي:** كل عنصر يحتاج إلى مساحة إضافية لتخزين المرجع إلى العنصر التالي. هذا يعني أن القائمة يمكن أن تستخدم المزيد من الذاكرة لكل عنصر مقارنة بالمصفوفة.

**4. المرونة:**

* **المصفوفة (Array):**
  + **غير مرنة:** حجم المصفوفة ثابت، وعند الحاجة إلى إضافة أو إزالة عناصر، يتطلب الأمر إعادة تخصيص المصفوفة.
* **القائمة المترابطة (Linked List):**
  + **مرنة:** يمكن إضافة أو إزالة عناصر بسهولة في أي مكان داخل القائمة دون الحاجة إلى إعادة تخصيص الذاكرة.

**5. الاستخدامات:**

* **المصفوفة (Array):**
  + مثالية عند الحاجة إلى الوصول السريع للعناصر باستخدام فهرس ثابت أو عندما يكون الحجم معروفًا مسبقًا.
  + تستخدم في التطبيقات التي تحتاج إلى بيانات ثابتة ولا تتغير بشكل مستمر.
* **القائمة المترابطة (Linked List):**
  + مثالية عندما يكون حجم البيانات غير معروف أو عندما تكون العمليات مثل الإضافة والحذف هي الأكثر تكرارًا.

**6. العيوب:**

* **المصفوفة (Array):**
  + **إضافة أو حذف العناصر** قد يكون بطيئًا ويتطلب إعادة تخصيص للذاكرة.
  + لا يمكن تغيير الحجم بسهولة.
* **القائمة المترابطة (Linked List):**
  + **الولوج العشوائي** بطيء مقارنة بالمصفوفات.