

مقارنة بين أنواع القوائم المرتبطة

مقدمة

، القائمة (Singly Linked List - DLL) هذا المستند يقارن بين ثلاثة أنواع من القوائم المرتبطة: القائمة المرتبطة الأحادية (Circular Linked List - CLL) ، والقائمة المرتبطة الدائرية (Doubly Linked List - DLL).

يتضمن الاستخدامات، الإيجابيات، السلبيات، وملخصاً سريعاً.

النوع	السلبيات الإيجابيات الاستخدامات	السلبيات الإيجابيات الاستخدامات
-تنفيذ الشناك والطابور. القائمة المرتبطة الأحادية (SLL)	<ul style="list-style-type: none">• تخفيص ذاكرة ديناميكي بسيط.• مفيدة عند الحاجة للتنقل الأمامي فقط. - بسيطة في التنفيذ.• أقل استخدام لذاكرة لكل عقدة (مؤشر واحد فقط).• فعالة للتنقل المتسلسل. - لا يمكن التنقل للخلف.• البحث عن العقدة السابقة بطيء.• الإدراج أو الحذف في النهاية يحتاج للانتقال من الرأس .	<ul style="list-style-type: none">• مفيدة في التطبيقات التي تحتاج للتنقل الدائري مثل الجدولة الدورانية.• الألعاب متعددة اللاعبين أو قوائم التشغيل المتكررة. - يمكن التنقل في القائمة بالكامل من أي عقدة.• لا يوجد قيمة null في النهاية، مما يسهل التكرار.• فعالة للطابور الدائري. - أكثر تعقيداً قليلاً في التنفيذ.• يمكن الوقوع في حلقة لا نهائية إذا لم يتم الحذر.• أصعب في تصحيح الأخطاء مقارنة بالقائمة الأحادية .
-القائمة المرتبطة الدائرية - (CLL)	<ul style="list-style-type: none">• عند الحاجة للتنقل في كلا الاتجاهين. - يمكن التنقل في كلا الاتجاهين.• يسهل الإدراج والحذف في أي موقع.• أكثر مرونة من القائمة الأحادية. - تحتاج لمزيد من الذاكرة لكل عقدة (مؤشران).• تنفيذها أكثر تعقيداً.• يلزم الحذر عند تحديث المؤشرات لتجنب تسرب الذاكرة .	<ul style="list-style-type: none">••••••

ملخص

- **القائمة المرتبطة الأحادية**: بسيطة، تنقل أمامي فقط، استخدام ذاكرة منخفض.
- **القائمة المرتبطة الدائرية**: يسمح بالتنقل الدائري من أي عقدة، مثالية للمهام الدورية.
- **القائمة المرتبطة الثانية**: يسمح بالتنقل ذهاباً وإياباً، مرونة عالية في الإدراج والحذف.