

مقارنة بين أنواع القوائم المرتبطة

مقدمة

، القائمة (Singly Linked List - SLL) هذا المستند يقارن بين ثلاثة أنواع من القوائم المرتبطة: القائمة المرتبطة الأحادية (Doubly Linked List - DLL)، والقائمة المرتبطة الثنائية (Circular Linked List - CLL) المرتبطة الدائرية. يتضمن الاستخدامات، الإيجابيات، السلبيات، وملخصًا سريعًا.

السلبيات الإيجابيات الاستخدامات النوع

-تنفيذ الستاك والطابور. القائمة المرتبطة الأحادية (SLL)

- تخصيص ذاكرة ديناميكي بسيط.
- مفيدة عند الحاجة للتنقل الأمامي فقط. | - بسيطة في التنفيذ.
- أقل استخدام للذاكرة 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 | .
- القائمة المرتبطة الدائرية - | (CLL) مفيدة في التطبيقات التي تحتاج للتنقل الدائري مثل الجدولة الدورية.
- 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 تعقيدًا قليلًا في التنفيذ.
- يمكن الوقوع في حلقة لا نهائية إذا لم يتم الحذر.
- أصعب في تصحيح الأخطاء مقارنة بالقائمة الأحادية | .
- القائمة المرتبطة الثنائية - | (DLL) تنفيذ deque، والتنقل (أمامًا وخلفًا) في التطبيقات مثل المتصفحات أو ميزة التراجع.
- 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 (مؤشرين).
- تنفيذها أكثر تعقيدًا.
- يلزم الحذر عند تحديث المؤشرات لتجنب تسرب الذاكرة | .

ملخص

- القائمة المرتبطة الأحادية: بسيطة، تنتقل أمامي فقط، استخدام ذاكرة منخفض.
- القائمة المرتبطة الدائرية: يسمح بالتنقل الدائري من أي عقدة، مثالية للمهام الدورية.
- القائمة المرتبطة الثنائية: يسمح بالتنقل ذهابًا وإيابًا، مرونة عالية في الإدراج والحذف.