



مقدمة عن البرمجة التفرعية

د. روان قرعوني

بسم الله الرحمن الرحيم

نبدأ وياكم هذا الفصل مادة البرمجة التفرعية التي ستكون مختلفة عن طريقة التفكير في البرمجة التقليدية التي تعلمناها مسبقاً فدعوا الله أن يوفقنا لهذا العمل

البرمجة التفرعية

في البرمجة التقليدية نقوم بكتابة Code ينفذ بشكل متسلسل، ولكن في البرامج الكبيرة والتي تحتاج إلى زمن تنفيذ كبير، نكون بحاجة للحصول على النتائج بسرعة أكبر، ومن هنا ظهرت الحاجة إلى البرمجة التفرعية حيث يتم استخدام أكثر من معالج على التوازي لضمان السرعة والتي سنتطرق للاستفاضة عنها هذا ضمن المحاضرات

ما الفرق بين البرمجة التفرعية و Threads؟

عندما نقوم بإنشاء 2 threads فإن التنفيذ سيكون جزء من الـ thread الأول يليه جزء من الـ thread الثاني ثم نكرر (أي أن تنفيذ التعليمات لا يتم في الوقت نفسه كما يظهر لنا)
بينما في البرمجة التفرعية تكون العملية مختلفة حيث ان التنفيذ يكون في الوقت نفسه على التوازي



فهدفنا من تعلم واستخدام البرمجة التفرعية هو استغلال قدرة الحواسيب المتاحة على التوازي بشكل كامل لإنجاز المهام بأسرع وأقل وقت

إضافة إلى ذلك أنه يوجد العديد من المشاكل في الـ Threads حيث تكون صعوبة الإدارة خاصة وإنها تشترك بذاكرة واحدة وهذا ما يسبب مشكلة إدارة الذاكرة

وهنا فإن البرمجة التفرعية تقدم حلول لبعض هذه المشاكل حيث نخصص لكل معالج ذاكرة خاصة أما في Threads فتكون تتشارك نفس المعالج والذاكرة نفسها.

مثال توضيحي:

ليكن لدينا مصفوفة مربعة، ونريد ضرب جميع عناصرها بـ 2 .

في البرمجة التقليدية : نستخدم حلقتي for للمرور على جميع العناصر وإجراء التعديل المطلوب

أما في البرمجة التفرعية : نقوم بتقسيم المصفوفة إلى أقسام، نوزع الأقسام على المعالجات المتاحة، وكل معالج ينفذ الجزء الموزع له، ثم نقوم بتجميع النتائج من المعالجات للحصول على النتيجة النهائية

■ الآن لنفرض أن زمن تنفيذ المثال السابق في البرمجة التقليدية كان 16، وعند تطبيق مفهوم البرمجة التفرعية على نفس المثال تم توزيع العمل بالتساوي على 4 معالجات، هل يكون زمن التنفيذ في البرمجة التفرعية هو 4 ؟

■ لا، لأنه يوجد على الأقل:

1. زمن توزيع

2. زمن تجميع

■ كما أنه يمكن مواجهة مشاكل في توزيع العمل على المعالجات،

في مثالنا يمكننا تقسيم المصفوفة إلى 4 أجزاء وتوزيعها على 4 معالجات، بينما في برامج أخرى قد يوجد ترابطات أي أن خرج العملية A هو دخل للعملية B وبالتالي لا يمكن تنفيذ التعليمة B على التوازي لأنها تتطلب الانتهاء من تنفيذ A أولاً في هذه الحالة نقول عن العمليتين A، B أنها غير قابلتين للتفرع

التطبيقات على استخدام البرمجة التفرعية:

معالجة صور الغلاف الجوي لتوقع الطقس حيث قد تصل أبعاد مثل هذه الصور إلى الملايين وتحتاج لمعالجتها لوقت كبير جداً لذلك نلجأ لتقسيم معالجتها لعدة أقسام ولعدة معالجات من أجل توفير الوقت المستغرق..



The End...

