

Drabiażgi

Do tej pory nie wiadomo dokładnie, jak należy organizować zespoły pracujące nad wielkimi programami. Jedną z propozycji jest tzw. zespół głównego programisty, którego struktura wzoruje się na zespole chirurgicznym, składającym się z wyspecjalizowanych fachowców (anestezjolog, instrumentariuszka, asystenci) pracujący pod bezpośrednim kierunkiem chirurga. Główny programista odpowiada, oczywiście, chirurgowi, a wspomagają go programiści — asystenci, wykonujący zlecane podprogramy, specjalista od dokumentacji, specjalista od testowania lub weryfikowania poprawności itp. Pierwsza próba takiego zespołu miała miejsce przy okazji komputeryzacji przez firmę IBM archiwum dziennika *New York Times*. Efekty były zachęcające, choć złośliwi twierdzą, iż do zespołu wzięto tak znakomitych programistów, że przy dowolnej organizacji zespołu efekty musiały być takie...

Do największych i najbardziej skomplikowanych wytworów myśli ludzkiej należą wielkie programy komputerowe, zwłaszcza te, które stanowią tzw. oprogramowanie podstawowe lub narzędziowe (systemy operacyjne, kompilatory języków programowania, systemy zarządzania bazami danych). Napisanie kompilatora języka Ada, opracowanego pod egidą Departamentu Obrony USA, wymaga co najmniej 100 osobo-lat pracy programistów, nie licząc pracowników obsługi, kosztów sprzętu, narzędzi programowych itp. System operacyjny OS 360/370 dla dużych komputerów firmy IBM wymagał, według różnych źródeł, od kilkudziesięciu do kilkuset tysięcy osobo-lat pracy. Zakładając, że w tak dużych programach wydajność statystycznego programisty wynosi do 30 wierszy programu dziennie, można oszacować rozmiar takich programów!

Kilkanaście lat temu Departament Obrony USA zlecił zaprojektowanie języka programowania, w którym można by napisać system operacyjny, a następnie w sposób formalny dowiedzieć jego poprawności. Chodziło o to, aby można pod kontrolą takiego systemu przetwarzać tajne dane równocześnie z nietajnymi i aby można całkowicie wiarygodnie dowiedzieć, że nikt niepowołany nie będzie mógł dostać się do tych pierwszych. W związku ze spadkiem cen sprzętu program zarzucono (wygodniej i pewniej było kupić osobny komputer do tajnych informacji), ale nawet gdyby się zakończył sukcesem, oficerowie kontrwywiadu nie mogliby spać spokojnie. Program, aby działał na komputerze, musiałby być przetłumaczony na język wewnętrzny, a kto potrafiłby zagwarantować poprawność programu tłumaczącego?

Przy analizie sygnałów elektronicznych (głównie radarowych, ale także będących wynikiem badań fizycznych i astronomicznych) korzysta się z transformacji Fouriera. Pozwala ona odtworzyć amplitudy i częstotliwości fal sinusoidalnych, których superpozycją jest badany sygnał. Taki sam algorytm pozwala zresztą potem odtworzyć sygnał na podstawie składowych. Klasyczny algorytm Fouriera wymaga rzędu $n \times n$ mnożeń liczb zespolonych (gdzie n jest liczbą zmierzonych lub wyliczonych wartości amplitudy sygnału). Trzydzieści lat temu odkryto tak zwany szybki algorytm transformacji Fouriera (FFT), wymagający tylko $n \times \lg(n)$ mnożeń. Załóżmy, że dysponujemy komputerem, który potrafi wykonać 100 tysięcy mnożeń liczb zespolonych na sekundę (typowa wartość dla bardzo dużych minikomputerów). Porównajmy: dla miliona wartości algorytm klasyczny wymagałby miliarda mnożeń, co trwałoby 10 milionów sekund (około 116 dni). Algorytm szybki wymagałby 20 milionów mnożeń, czyli około 200 sekund. Nic dziwnego, że odkrycie FFT stało się przełomowym momentem w analizie sygnałów, pozwalając na ogromne zwiększenie zasięgu i dokładności badań.

Komputery są bardzo intensywnymi źródłami ciepła. Zużywana (a więc i wydzielana) moc jest z grubsza proporcjonalna do szybkości działania. W wielkich komputerach moce dochodzą do setek kilowatów. Z drugiej strony, aby osiągnąć dużą szybkość działania, zmniejsza się rozmiar komputera, a więc całe to ciepło wydzielane jest w objętości około jednego metra sześciennego. Ciepło to trzeba odprowadzić na tyle szybko, aby temperatura wewnętrzna komputera nie przekroczyła około 70°C. Spełnienie tych warunków jest jednym z podstawowych problemów przy konstruowaniu superkomputerów. Stosuje się wymyślne układy chłodzące, podobne do szaf chłodniczych, a od kilku lat najszybsze komputery są zanurzane w beczkach wypełnionych płynem chłodzącym pod ciśnieniem. Z beczki wychodzą tylko rury doprowadzające i odprowadzające płyn chłodzący oraz przewody łączące układy elektroniczne z zasilaniem i z urządzeniami zewnętrznymi.

Płyta z zapisem laserowym (dysk kompaktowy) może być używana do przechowywania informacji cyfrowej. Mieści się wówczas na niej około 500 MB (megabajtów, czyli milionów znaków). Jest to pięć razy tyle, ile zawiera trzynastotomowa Wielka Encyklopedia Powszechna PWN. Ale z drugiej strony do zapamiętania obrazu składającego się z miliona punktów (1000 rzędów po 1000 punktów), z których każdy może mieć jeden z 64 kolorów (po cztery poziomy czerwieni, zieleni i błękitu), potrzeba 0,75 MB. Gdyby zapisać ruchomy film (50 obrazów na sekundę, tyle, co w telewizji), to na płycie zmieściłoby się tylko około 13 sekund. Oczywiście, w rzeczywistości ilość informacji przekazywana w czasie transmisji telewizyjnej jest znacznie mniejsza; obraz składa się z mniejszej liczby punktów, a sąsiednie punkty ekranu nie mogą mieć zupełnie dowolnych kolorów.

Jeszcze dziesięć lat temu w Centrum Ochrony Obszaru Powietrznego Ameryki Północnej (NORAD) pracowały dwa komputery lampowe, zainstalowane w latach pięćdziesiątych. Podobno przez przeszło dwadzieścia lat w każdej chwili co najmniej jeden z nich działał! Komputery zajmowały wielką salę w podziemnym schronie, a ich obsługa wymagała wymiany co tydzień 200 lamp elektronowych. Przez 20 lat żaden producent nie potrafił dostarczyć nowocześniejszego, a równie niezawodnego sprzętu. Dopiero na początku lat osiemdziesiątych udało się to firmie Hewlett-Packard; dwa zainstalowane nowe komputery zajmują mniej miejsca niż same klimatyzatory potrzebne starym. Wielka sala komputerowa pozostała pusta — pracownicy Centrum proponowali zrobić w niej boisko do siatkówki.

Z algebry liniowej wiadomo, że można rozwiązać każdy oznaczony układ n równań z n niewiadomymi za pomocą wyznaczników. Okazuje się jednak, że koszt obliczenia wyznacznika rośnie wykładniczo wraz ze wzrostem n . Co gorsza — przy skończonej dokładności obliczeń błąd obliczeń może być ogromny! Znacznie lepsze wyniki otrzymuje się stosując algorytm rugowania zmiennych (eliminacja Gaussa), którego koszt jest proporcjonalny do n^3 . W latach siedemdziesiątych udało się znaleźć inne, jeszcze trochę szybsze algorytmy (m. in. Strassena). Okazało się, że jeśli potrafimy pomnożyć dwie macierze $m \times m$ wykonując k mnożeń, to potrafimy rozwiązać układ równań kosztem n^a , gdzie $a = \log_m k$. Standardowa metoda mnożenia macierzy zawsze daje $a = 3$. Strassen pokazał, że można pomnożyć macierze 2×2 korzystając z siedmiu mnożeń, otrzymując $a = 2,8$. Aby poprawić ten wynik, należało znaleźć algorytm mnożenia macierzy 3×3 korzystający z co najwyżej 21 mnożeń, co też się wkrótce udało. Wprawdzie można w ten sposób tworzyć coraz lepsze algorytmy, ale uzyskana poprawa wyniku nie jest tak radykalna, jak to było np. w wyniku algorytmu FFT.

