Projekt Zespołowy
Etap projektu – projektowanie rozwiązania na zadaną architekturę

Abstract

Poniższe sprawozdanie jest wynikiem naszej pracy na drugim etapie projektu zespołowego z implementacji metody indeksu w architekturach GPU. Przedstawimy w nim przygotowane przez nas projekty i rysunki koncepcyjne wymaganych do zaimplementowania algorytmów.

1 Analiza możliwości implementacji algorytmów mnożenia modularnego dużych liczb

Zaproponowanym przez nas algorytmem jest ten odkryty przez rosyjskiego matematyka Anatolija Karacubę. Umożliwia on zmniejszenie złożoności czasowej ($\Theta(n^{log_2 3})$) w porównaniu do mnożenia klasycznego ($\Theta(n^2)$) .

Projekt algorytmu:

Do pomnożenia dwóch n-cyfrowych liczb x i y przy podstawie B, gdzie $n = 2m$ (jeśli n jest nieparzyste, albo x ma różną liczbę cyfr niż y, można to naprawić, dodając zera po lewej stronie tych liczb), rozpisujemy je jako:

$$x = x_1 B^m + x_2$$

$$y = y_1 B^m + y_2$$

gdzie x_2 i y_2 są mniejsze niż B^m. Wynik mnożenia wynosi wtedy:

$$xy = (x_1 B^m + x_2)(y_1 B^m + y_2) = x_1 y_1 B^{2m} + (x_1 y_2 + x_2 y_1) B^m + x_2 y 2$$

Standardową metodą byłoby pomnożenie czterech czynników osobno i dodanie ich po odpowiednim przesunięciu. Daje to algorytm działający w czasie $O(n^2)$. Karacuba zauważył, że możemy ten sam wynik uzyskać, wykonując tylko trzy mnożenia:

$$X = x_1 y_1$$

$$Y = x_2 y_2$$

$$Z = (x_1 + x_2)(y_1 + y_2) - X - Y$$

Dostajemy wtedy:

$$Z = (x_1y_1 + x_1y_2 + x_2y_1 + x_2y_2) - x_1y_1 - x_2y_2 = x_1y_2 + x_2y_1)$$

A zatem $xy = XB^{2m} + Y + ZB^m$; tym samym kosztem kilku dodawań i odejmowań zmniejszyliśmy liczbę mnożeń z czterech do trzech.
Każde z tych mnożeń m-cyfrowych liczb możemy znów wykonać za pomocą algorytmu Karacuby, wykorzystując rekurencję.

2 Analiza możliwości implementacji algorytmu poszukiwania relacji (oraz faktoryzacji w bazie), dla algorytmu metody indeksu

Wstęp do wykorzystywanego algorytmu: 1. Wyznaczamy wszystkie liczby pierwsze mniejsze od B, które tworzą bazę rozkładu. 2. Losujemy l liczb a_i, gdzie $i = \overline{1, l}$, takich że rozkładają się one w wyznaczonej wcześniej bazie rozkładu. 3. Dzielimy liczby a_i, gdzie $i = \overline{1, l}$ przez kolejne liczby z bazy rozkładu dopóki wynik dzielenia jest całkowitoliczbowy.

3 Analiza możliwości implementacji algorytmu eliminacji Gaussa nad ciałem $\mathbb{F}_p$, dla ciał o dowolnym rozmiarze