

## Описание решения

Мой номер - 8, следовательно при выполнении данного задания мне досталась конструкция `_mm_and_pd`. Функция `_mm_and_pd` в C++ обычно используется с Intel Intrinsics для программирования SIMD (одна инструкция, несколько данных). Она выполняет побитовую операцию AND над упакованными значениями с плавающей запятой двойной точности.

### main.cpp

Единственный модуль, эта программа использует функцию `_mm_loadu_pd` для загрузки значений из обычных массивов в регистры SIMD, выполняет поразрядную операцию AND, используя `_mm_and_pd`, а затем сохраняет результат обратно в обычный массив, используя `_mm_storeu_pd`. Тип `__m128d` представляет 128-битный SIMD-регистр, который может хранить два значения двойной точности с плавающей запятой.

### Makefile

Файл для сборки описанного проекта. Для запуска необходимо выполнить в терминале следующие команды:

1. `make`
2. `./intrinsik_taskk`

Также можно чистить директорию от объектников, выполнив `make clean`.

### Заключение

Очевидно, что данный способ поразрядной битовой операции быстрее наивной реализации и может быть применен в более сложных конструкциях, где надо получить результат применения битового AND.