В качестве реализации третьего задания предлагается расширить полученное 2е задание поддержкой GPU – каждый процесс должен использовать как минимум один GPU. Для успешной сдачи задания вводятся следующие требования:

1) Срок **начала** сдачи программы – **10 декабря,** срок **окончания приема** сдачи – **15 декабря**
2) Программа должна быть на MPI + CUDA.
3) Программа должна собираться через makefile, в котором обязательно должны быть две переменные **ARCH**=sm_N (N = 35 / 60), обозначающая архитектуру GPU, и **HOST_COMP=mpicc,** обозначающая хост компилятор. Эти переменные должны использоваться как минимум для **nvcc.** Запрещается использовать возможности CUDA > cc 3.5.
4) Отчет о выполнении должен содержать в себе этап выполнения 2 и 3, то есть распараллеливание на MPI, MPI + OpenMP и MPI + CUDA.
5) В отчете должны содержаться все времена запусков задачи на том количестве процессоров, которое требуется. Также должны быть получены графики ускорения и эффективности, по отношению к **последовательной** (исходной!) программе без MPI / OpenMP.
Опционально можно посчитать ускорения различных параллельных версий между собой.
6) Программа на MPI + CUDA должна работать **НЕ медленнее**, чем MPI, и MPI + OpenMP. Если количество данных не хватает для получения «хороших» цифр на GPU, следует увеличить исходные размеры массивов.
7) Отчет должен содержать пояснение по тем результатам, которые были получены – каков характер ускорений, эффективности, каковы причины такого поведения, если ожидаемые цифры не совпадают с реальными.
8) В отчете должно быть указано, каким образом производилась оценка корректности выполнения параллельных версий, в особенности MPI + CUDA.
9) В отчете должны содержаться не только общее время работы программы, а также времена всех параллельных циклов, времена инициализации и завершения работы программы, времена копирования данных с GPU на хост и обратно, времена коммуникационных обменов (если они асинхронные, то демонстрация того, что они не занимают времени) как в исходной, последовательной программе, так и в параллельной. Таким образом, таблица, содержащая времена запусков, должна содержать помимо общего времени, времена всех затрат на коммуникации и обмены между GPU, а также времена параллельных циклов.
Данные подробные оценки времен можно приводить только для одного запуска, например, с самыми большими входными данными, чтобы продемонстрировать хорошее ускорение.
10) Для сдачи необходимо прислать исходный код программы и готовый отчет по 2 и 3 заданиям. Исходный код должен содержать исходную (не испорченную) версию программы, и параллельную.
11) **Запрещается** использование разделяемой памяти для реализации редукции. Использование разделяемой памяти где-либо требует обоснования в отчете.
12) Предпочтительнее использовать thrust.

Невыполнение **хотя бы одного** из этих пунктов приведет к дополнительной итерации сдачи 3го задания по реализации не выполненных пунктов.