

**0) Задание состоит в том, чтобы расширить имеющуюся программу в 3м задании с использованием MPI+CUDA/MPI+OpenACC/CDVMH.**

1) Срок **начала** сдачи программы – не позднее **7 декабря**, срок **окончания** сдачи – **20 декабря**, далее задачи будут приниматься только в модели DVMH.

2) Программа должна быть на MPI + CUDA, MPI+OpenACC или в модели CDVMH. Использование библиотек cuBLAS/cuSPARSE запрещается.

3) Программа должна собираться через makefile, в котором обязательно должны быть две переменные **ARCH=sm\_N** (N = 35 / 60), обозначающая архитектуру ГПУ, 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 все предыдущие этапы задания, то есть распараллеливание на MPI, OpenMP, а также MPI + GPU.

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 + GPU должна работать **НЕ медленнее**, чем MPI, OpenMP. Если количество данных не хватает для получения «хороших» цифр на ГПУ, 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 прогнозируемыми.

8) В отчете должно быть указано, каким образом производилась оценка корректности выполнения параллельных версий, в особенности MPI + GPU.

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 ГПУ 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 ГПУ, а также времена параллельных циклов.

10) **Запрещается** использование разделяемой памяти для реализации редукции. Использование разделяемой памяти где-либо требует обоснования в отчете.

11) **Для сдачи необходимо загрузить код в любой git-репозиторий** и прислать ссылку на этот репозиторий **в личном сообщении**. В данном репозитории желательно иметь историю коммитов и изменений в процессе распараллеливания задачи, а также исходный код программы (в разных ветках последовательный и параллельный) и готовый отчет.

Невыполнение **хотя бы одного** из этих пунктов приведет к дополнительной итерации сдачи 4го задания для реализации не выполненных пунктов.

**PS: Производительность IBM Power 8 DP – 0.3 Tflops, Tesla P100 – 4.7 Tflops**

Скорость памяти IBM Power 8 – for 1 TB RAM – 230 GB/s, Tesla P100 – 700 GB/s

**Из этих соотношений ясно, что для ГПУ может быть получена достаточно хорошая программа при условии оптимальности кода.**