CME213 PA3

Dong-Bang Tsai

(1) I can allocate memory using thrust template container like what we do in C++ STL. The following example is to allocate 100 int array in memory using container in device and host.

```
thrust::host_vector<int> host_vector_container(100);
thrust::device_vector<int> device_vector_container(100);
```

Of course, we can allocate the memory manually using thrust; however, we have responsibility to deallocate with device free. The following is short example.

```
thrust::device_ptr<int> device_vector_array = thrust::device_malloc<int>(100);
thrust::device_free(host_vector_array);
```

One useful feature is that we can cast pointer of template container to raw pointer which will be very useful. Since the container is host code, which can not be called in device kernel function, the only way to pass the array directly into device kernel is the raw pointer.

```
thrust::device_ptr<int> host_vector_array
        = thrust::raw_pointer_cast(&device_vector_container[0]);
```

(2) I have the images in the next pages. There are two images, and the one on the left is the frequency table for cipher_text.txt encrypted by 500 length key. You can see that the distribution is almost uniform across all the letters and digraphs.

However, the one on the right is generated by using plain_text.txt. You can see that e is the most likely letter, and for the bigraphs, there are several recognized patterns.

(3) I used transform, transform_iterator, remove_copy_if, counting_iterator, sort, upper_bound, sort_by_key, reduce, and inner_product.

(4) Since the code can not be run with -G flag, here, I mainly compared the performance between -O3 and -g flags. As the following output, the release version with -O3 version is 35% faster than the debugging version.

```
[dbtsai@node006 programming]$ make
nvcc -o create_cipher create_cipher.cu -O3 -arch=sm_20
nvcc -o solve_cipher solve_cipher.cu -O3 -arch=sm_20
[dbtsai@node006 programming]$ time for i in {1..10}; do ./solve_cipher cipher_text.txt > /dev/null; done
real    0m17.005s
user    0m3.460s
sys     0m13.115s

[dbtsai@node006 programming]$ make clean && make DEBUG=1
rm create_cipher solve_cipher
nvcc -o create_cipher create_cipher.cu -arch=sm_20 -g -G
nvcc -o solve_cipher solve_cipher.cu -arch=sm_20 -g -G
[dbtsai@node001 programming]$ time for i in {1..10}; do ./solve_cipher cipher_text.txt > /dev/null; done

real    0m27.812s
user    0m4.970s
sys     0m21.568s
```

(5) By adding the text repeated several times over to the plain text, for example, convert the string “The quick brow fox jumps over the lazy dog” into “thequickbrowfoxjumpsoverthelazydogthequickbrowfoxjumpsoverthelazydog thequickbrowfoxjumpsoverthelazydog”, there will be two periodicity involved, which will change the behavior of ioc, and it will cause the solver failed.