NTU, CSIE

# GPGPU Programming Lab 1

江東峻 B01902032

March 28, 2016

## 1 Count the Position in Words

## 2 Find the Heads

## 3 Part3: Reverse the Substrings

In this part, we try to reverse each substring.

| **Sample Outputs for Part 3** |
| --- |
| abc a ab abcd (input) |
| cba a ba dcba (output) |

We seperate this task into 2 segments:

1. Find the heads and tails of substrings.
2. Swap the characters in each substring.

### 3.1 Find the heads and tails of substrings

In this segment, we define the binary function $is\_continuous()$ and use $thrust::unique\_by\_key()$ to get the discontinuous points (including heads and tails). The array $head\_tail$ looks like [$head1$, $tail1$, $head2$, $tail2$, ...]. Notice that we need to add a "final" tail in the end if the length of $head\_tail$ is odd.
Now we have the heads and tails of all substrings. In the next segment, we will label each character in the whole string with the index of the substring to which it belongs.

### 3.2 SWAP THE CHARACTERS IN EACH SUBSTRING

To label each character, we define a kernel function $headTagging()$ and label the head indices first. Here we change the label of the first head into 0 such that the labels of substring start from 0. Then, we use $thrust::inclusive\_scan()$ to get all labels.
Now we define the second kernel function $reverseEachString()$ to swap the each corresponding pair of characters in each substring. Notice that if the "index_in_string" exceeds half of the substring length, we will do nothing. (to avoid swapping twice in the same pair)

| **Sample Structures** |
| --- |
| abc a ab abcd (input) |
| 0, 3, 4, 5, 7, 9, 10, 14 (head_tail) |
| 00001001001000 (headTagging, the first head is labeled with 0) |
| 00001112223333 (inclusive_scan) |