Q5

```
int intput_sum[N];
prefix_sum(N, input, input_sum);
int index[N];
prefix_sum(N, seq_starts, index);

int sub_len[4];
int sub_sum[4];
get_len_and_sum<<<N>>>(N, input_sum, index, sub_len, sub_sum);

int result[4];
for (int i = 0; i < 4; i++) {
  if (i == 0) result[i] = sub_sum[i] / sub_len[i];
  else {
    result[i] = (sub_sum[i] - sub_sum[i - 1]) / (sub_len[i] - sub_len[i - 1]);
  }
}
return result;


/* kernel function */
void get_len_and_sum(int N, int *input_sum, int *index,
                     int *sub_len, int *sub_sum) {
  int index_in_sub = 0;
  if (current_id == N - 1) index_in_sub = 3;
  else if (index[current_id + 1] != 0) index_in_sub = index[current_id + 1] - 2;
  sub_len[index_in_sub] = current_id + 1;
  sub_sum[index_in_sub] = input_sum[current_id];
}
```