

# KẾ HOẠCH LÀM VIỆC

~~10/01/2020~~

## Thành viên

Huỳnh Minh Huân	1612858
Nguyễn Ngọc Khải	1612909

## Chi tiết

- Nhóm sử dụng k mặc định là 8 (xét 8 bit) để đo thời gian thực thi.
- Đối với base03, base04\_v3, base04\_v4, base04\_v5: tính thời gian thực hiện trung bình (số lần lặp mặc định là 16).

	thời gian	thời gian thực thi (ms)	công đoạn/ghi chú
Thuật toán Radix Sort tuần tự ( <b>base01</b> )	16/12/2019	1160	Đã thực hiện ở bài tập số 4.
Song song 2 bước hist và scan ( <b>base02</b> )	16/12/2019	776	Đã thực hiện ở bài tập số 4.
Thuật toán Radix Sort song song với k = 1 ( <b>base03</b> )	20/12/2019	179	
Thuật toán Radix Sort tuần tự theo hướng dẫn đồ án ( <b>base04_v1</b> )	24/12/2019	2119	Cài đặt mảng lưu bin như file hướng dẫn. (hàng là các block, cột là các bin)
Song song 2 bước tính hist và scan ( <b>base04_v2</b> )	26/12/2019	1584	Thay đổi cách lưu mảng (hàng là các bin, cột là các block) -> scan dễ dàng hơn. Tính histogram trên global memory.
Cài đặt preScatter và scatter song song ( <b>base04_v3</b> )	29/12/2019	197	Tính histogram trên SMEM. tối ưu kernel scan. Huân (pre_Scatter, tối ưu), Khải (scatter).
Cài Scatter song song ( <b>base04_v4</b> )	02/01/2020	149	gộp 2 kernel ở ver3 thành 1 kernel
<b>base04_v5</b>	03/01/2020	138	

Thuật toán Radix sort do thrust hỗ trợ		41	
--	--	----	--

## Kết quả

- Thuật toán cho kết quả tốt nhất với:  $k = 4$ , blockSize1 hist là 128, blockSize2 là 1024.
- Thời gian trung bình là 63 (ms) so với thời gian thực hiện của Thrust là 41 (ms).

```

*****GPU info*****
Name: GeForce GTX 1080 Ti
Compute capability: 6.1
Num SMs: 28
Max num threads per SM: 2048
Max num warps per SM: 64
GMEM: 11713052672 byte
SMEM per SM: 98304 byte
SMEM per block: 49152 byte
*****

Input size: 16777217
nBits: 4

Hist block size: 128, scan block size: 1024

Radix sort by host
Time: 2281.008 ms

Radix sort by host level 1
Time: 3727.513 ms
CORRECT :)

Radix sort by device level 2
Avg Time: 63.894 ms
CORRECT :)

Sort by thrust
Time: 41.541 ms
CORRECT :)

```

## Tham Khảo

[Using Shared Memory in CUDA C/C++](#)