Advanced Algorithms Programming Assignment #2

Find the closest pair of points in 1D, 2D, and 3D.

Due date: **2022.11.21 09:00 AM KST**

- **Submit via LearnUs**

In this assignment, you are asked to find the closest pair of points in 1D, 2D, and 3D coordinate space.s

Instruction

-Implementation

- Implement kernel.c and kernel.h if necessary.
- Compile the project using make command.
- You can test your program with your own input using ./closest-pair <n|daq> <1d|2d|3d> command.

-Competition

- Competition will be based on finding the closest pair of points in 2D space by divide and conquer algorithm. Number of points will be 99999
- make competition command will be used for competition.
- Obviously, your program must correctly find the closest pair in the first place.

Details

- Details of finding closest pair of points problem :

  https://en.wikipedia.org/wiki/Closest_pair_of_points_problem .

Requirements

- Implementation MUST be in C
- Include necessary comments for understanding.
- Your implementations MUST work in Ubuntu 18.04 / 20.04 (This is what the TA will use for grading)
- Follow the skeleton code structure.
- Do not use 3rd party libraries.

Report (in PDF format)

- Your report must follow common technical writing guidelines which includes the following details. Lack of each detail will result in **1pt deduction.**
  - Summary/Abstract
  - Explanation about your code. (Focus on big pictures, not the details)
  - Results and Discussion **with Graphs and Tables. (No naive screenshots of the result is allowed.)**
  - Takeaways
    - What have you learned?
    - What was easy? What was your huddle and how did you overcome it?
  - Feedback

        - Feedback for the assignment.
- Your report could be either written in Korean or English.
- **Please submit your report in 'PDF' format!!**

Grading

- Zip your submissions in *StudentID_Name.zip (Ex : 2022314412_YujinShin.zip)*
- Name your report as *StudentID_Name.pdf (EX : 2022314412_YujinShin.pdf)*
- 0 pts if your program does not compile, results in segmentation fault, etc.
- Total 15pts deduction if you do not follow the skeleton code structure(5pts each for 1D, 2D, 3D).
- Details
    - 1D (30 points)
    - 2D (30 points)
    - 3D (30 points)
    - Report (10 points)

[Optional] Extra credit task

- Implemetation in other languages such as Rust, C# will be 10 pts for each 1D, 2D and 3D implementation.

[Optional] Top 3: Speed competition

- Speed competition of finding the closest pair of points in 2D space by divide and conquer algorithm.
- Bonus point policy: 30 pts for 1st place, 20pts for 2nd place, 10pts for 3rd place.

- Measurement will be done on TA's ubuntu PC.
- While your final(HW) submission will be used for the real competition, you have at most 2 chances to contact TA for practice submission. Please email raki2001@yonsei.ac.kr following the format below.
    - Title : [Advanced Algorithms] 홍길동 PA2 2회째 시도입니다.
    - Attachment : kernel.c, kernel.h (if changed)
- Result will be uploaded on the Slack.