Names: Peyton Kelly, Gary Chen, William Connelly

For our final project in the class we look to tackle the seven sort option. To develop a platform that executes and compares the performance of seven different sorting algorithms—bubble, insertion, selection, counting, quick, merge, and radix—using Polish names and randomly generated salary data as input. This platform will enable users to understand the strengths and weaknesses of each algorithm and make informed decisions when implementing sorting solutions.Sorting algorithms are fundamental to computer science and have widespread applications across various fields, such as data analysis, search engines, and databases. The choice of the right sorting algorithm depends on the type, size, and distribution of the data. By developing a platform that can efficiently execute multiple sorting algorithms, users can make better-informed decisions on which algorithms best suit their needs.

The real problem we aim to tackle is the lack of a comprehensive platform that allows users to understand and compare the performance of various sorting algorithms using real-world data. We plan to solve this computing problem by developing the SACP, implementing the seven sorting algorithms, creating a dataset of Polish names and salary information, and providing a user-friendly interface for selecting algorithms, input data type, and data size.The primary objective is to enable users to understand the strengths and weaknesses of each algorithm and make informed decisions when implementing sorting solutions.

References:

1. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press.
2. Knuth, D. E. (1998). The Art of Computer Programming, Volume 3: Sorting and Searching (2nd ed.). Addison-Wesley.
3. Sedgewick, R., & Wayne, K. (2011). Algorithms (4th ed.). Addison-Wesley.