Assignment#2: External Sort with Replacement Selection Algorithm

Due date: 15th June, 23:59

Goal

In this assignment you are asked to write a computer program in C that reads a given text file and sorts it using the Sort-Merge approach and the Replacement Selection Sort to order the initial sorted segment. The program should be a console application and accept command line arguments as follows:

```
> XSort unsortedData.txt sortedData.Txt B=5 P=32
Sort operation will take 3 phases.
Phase 0: 8 sorted segment created initially
Phase 1: 2 merged segments created
Phase 2: Final sorted file has been created.
```

The name of the program is XSort and and should accept three parameters. The first one is the name of the input file which contains the data to be sorted. The second one is the name of the output file where sorted data to be stored. The next, *B*, is the number of buffers to be used in Merge phases. The last one, *P*, is the page size in KB. The program should also give some information about the phases which are executed. You can use the structure given below in your coding. The sorting should be done by taking the "id" parameter of the record into account.

```
struct record
{
    int id;
    int grade;
   char name[15];
   char surname[15];
   char email[26];
};
```

Implementation Requirements

- The assignment must be coded in C.
- You must write your code according to Replacement Selection Sort rules.
- You should input Disk Page Size and Number of Buffers dynamically as an argument.

Documentation

In this assignment, in line documentation is expected, as well as good coding practices such as consistent naming, proper usage of indentation and high readability of code.

Submission

- Name your source code file *xxx*.c, where *xxx* is your ***student id***. If you don't follow the naming rules, a penalty applies. (10 pts)
- Late submission is accepted but, 10 points penalty applies for each day.

Honesty

Your submissions will be scanned among each other as well as the Internet repository. Any assignments that are over the similarity threshold of a system for Detecting Software Similarity will get zero. We strongly encourage you not to submit your assignment rather than a dishonest submission.

Grading policy

- File reading - %25
- Replacement Selection Sort - %25
- Merge-Sort - %25
- Writing Sorted File - %25

For Questions

For any questions about the assignment please write under the topic "Homework2 Questions" in the Forum on the SAKAI platform. Before asking your question, please check carefully previous questions and answers, where similar questions that were asked by someone else were already answered.

- No private questions via email will be answered!!!
- We will try to answer any of your questions as soon as possible, except the ones "Hocam my code does not work, can you fix it" or "I have implemented it but it does not work, can you look at it". Debuggers are far more suitable options.

Good luck!!!

Read all of the instructions carefully, if you find something UNCLEAR, please ask help to CLARIFY it!