# Homework: Join Algorithms (w8)

New Attempt

**Due** May 31 by 11:59pm **Points** 8 **Submitting** a file upload

## Introduction

The objective of this assignment is to learn join implementation algorithms for relations and files on external memories.

## What you must do

Consider the following relations:

```
Dept (did (integer), dname (string), budget (double), managerid (integer))
Emp (eid (integer), ename (string), age (integer), salary (double))
```

Fields of types integer, double, and string occupy 4, 8, and 40 bytes, respectively. Each block can fit at most one tuple of an input relation. Using the provided skeleton code with this assignment, implement the optimized sort-merge join algorithm for Dept ◁▷Dept.managerid=Emp.eid Emp in C++.

- Each input relation is stored in a separate CSV file, i.e., each tuple is in a separate line and fields of each record are separated by commas. The files that store relations Dept and Emp are Dept.csv and Emp.csv, respectively. Your program must assume that the input files are in the current working directory, i.e., the one from which your program is running. We have included sample input CSV files with this assignment as sample test cases for your program. Your program must join correctly other CSV files with the same fields as the sample files.
- The program must store the result in a new CSV file with the name join.csv in the current working directory.
- During the join computation, your program may keep up to 22 blocks plus a relatively small number of control variables, e.g., flags or counters, in main memory at any time. The submitted solutions that use more main memory will not get any points.

- Each student has an account on hadoop-master.engr.oregonstate.edu server, which is a Linux machine. Your should ensure that your program can be compiled and run on this machine. You can use the following bash command to connect to it:

```
> ssh your_onid_username@hadoop-master.engr.oregonstate.edu
```

  Then it asks for your ONID password and probably one another question. To access this server, you must be on campus or connect to the Oregon State VPN.

- You must name the file that contains the source code of the main() function main5.cpp. If you place your source code in multiple files, you may submit all of them in a single zip file.

- You can use following commands to compile and run C++ code:

```
> g++ -std=c++11 main5.cpp -o main5.out
> main5.out
```

There is no need to use double buffering in your implementation.

## Necessary files

Input files:

**Dept.csv (https://canvas.oregonstate.edu/courses/1939345/files/99252269?wrap=1)** ⬇
**(https://canvas.oregonstate.edu/courses/1939345/files/99252269/download?download_frd=1)**

**Emp.csv (https://canvas.oregonstate.edu/courses/1939345/files/99252273?wrap=1)** ⬇
**(https://canvas.oregonstate.edu/courses/1939345/files/99252273/download?download_frd=1)**

**(https://canvas.oregonstate.edu/courses/1939345/files/99252273?wrap=1)** Following files contain the skeleton code to generate the required **join.csv** file. You may make changes to these files adhering to the assignment requirements.

**main.cpp (https://canvas.oregonstate.edu/courses/1939345/files/99252304?wrap=1)** ⬇
**(https://canvas.oregonstate.edu/courses/1939345/files/99252304/download?download_frd=1)**

**record_class.h (https://canvas.oregonstate.edu/courses/1939345/files/99252307?wrap=1)** 
**(https://canvas.oregonstate.edu/courses/1939345/files/99252307/download?download_frd=1)**

## What to turn in

The assignment is to be turned in before Midnight (by 11:59pm). You may turn in the source code of your program through Canvas. The assignment may be done in groups of two students. Each group may submit only one file that contains the full name, OSU email, and ONID of every member of the group.

## Grading criteria

The programs that implement the correct algorithm, return correct answers, and do not use more than allowed buffers will get the perfect score. The ones that use more buffer than allowed will not get any points.  The ones that implement the right algorithm but return partially correct answers will get partial credits.