

# **HW4 Report**

By Alexander Rosati

April 15, 2020

[Click here to watch the demo](#)

## Basic Information

<b>Student Name</b>	Alexander Rosati
<b>Language</b>	C#
<b>Technologies</b>	WPF/XAML, Visual Studios 2017
<b>GitHub Repository</b>	<a href="https://github.com/AlexanderRosati/HashJoinAlg-DBMS-Course">https://github.com/AlexanderRosati/HashJoinAlg-DBMS-Course</a>
<b>Email</b>	ajrosati@umich.edu
<b>Demo (~11 mins)</b>	<a href="https://youtu.be/HSepEsLroNE">https://youtu.be/HSepEsLroNE</a>

*Note: Go to the GitHub repository to download all of the source code.*

## UML Class Diagrams

<b>Index</b>
+ relationName:String + attribute:String - pathToTable:String - hashTable:List<Tuple<int, long>>[] - numberBuckets:const int
+ Index(rel:String, attr:String, path:String, fileForTable:String[]):Void - HashFn(valueOfAttribute:int):int + Search(valueOfAttribute:int):List<long>

<b>InnerTable</b>
+ indexes:Index[] + pathToFile:String + innerTableFile:String[] + numRows:int + relationName:String
+ InnerTable(path:String, name:String):Void

<b>OuterTable</b>
+ outerTableFile:String[] + pathToFile:String + numRows:int + relationName:String
+ OuterTable(path:String, relName:String):Void + HashJoin(innerTable:InnerTable, outerAttribute:String, innerAttribute:String):String

<b>TableManager</b>
+ outerTables:List<OuterTable> + innerTables:List<InnerTable>
+ TableManager(R1Dir:String, R2Dir:String):Void + Join(R1FileName:Str, R2FileName:Str, R1Attribute:Str, R2Attribute:Str):String + SelectivityOfJoin(R1FileName:Str, R2FileName:Str, R1Attribute:Str, R2Attribute:Str):Double

## Notes on Implementation

- Written in C#.
- WPF/XAML was used.
- Each instance of R2 has 4 indexes, one for each attribute. Instances of R1 have no indexes.
- R1 is always outer table and R2 is always inner table.
- Indexes of R2 are used to find matching records.
- Each instance of R1 and R2 was its own CSV file. E.g.,

### **R1-INSTANCE-1.csv**

```
a1,a2,a3
1,0,0
2,0,0
3,0,0
4,0,0
```

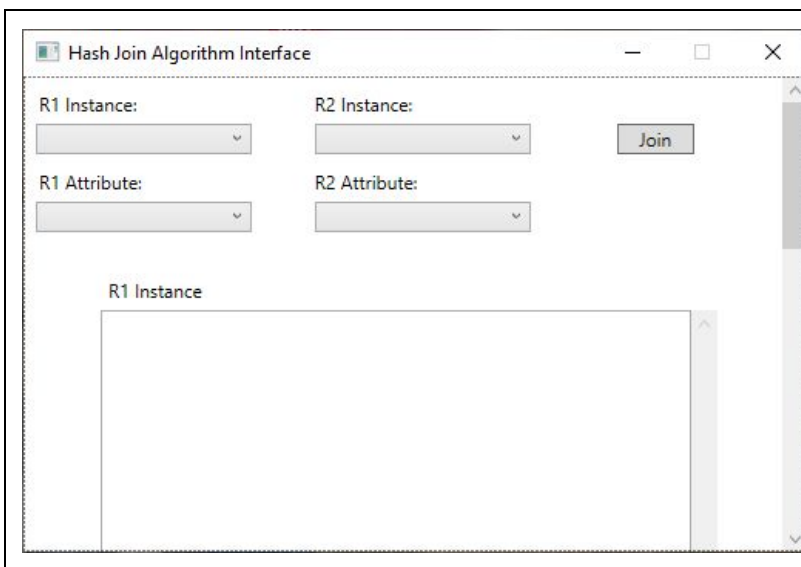
- “R1-INSTANCE-1.csv” is the file name. The file represents an instance of R1. “a1,a2,a3” are the attributes. Every line below is a record.

- There are 13 buckets for every hash table. 13 was chosen because it is a prime number. Prime numbers prevent clustering.
- Each file for an instance of R1 or R2 is read into memory and kept in memory until the program terminates. This will work for this application but would not scale well for obvious reasons.
- Hash table is an array of linked lists. Linked lists contain tuples.
- R1 Instance drop down box contains files that are in 'R1-INSTANCES' folder.
- R2 Instances drop down box contains files that are in 'R2-INSTANCES' folder.
- If you place a file in 'R1-INSTANCES' folder, it will appear in R1 Instance drop down box. File must be CSV.
- If you place a file in 'R2-INSTANCES' folder, it will appear in R2 Instance drop down box. File must be CSV.

## Program Usage

Go to the GitHub [repository](#) and download the source code. Unzip the folder. Go to 'HashJoinAlg-DBMS-Course-master' > 'HashJoinAlg-DBMS-Course-master' > 'HashJoinInterface' and open the solution. Right click the solution and then click 'Build Solution'. Close Visual Studios (which I assume you have on your machine). In the directory that contains the solution, click on the folder 'HashJoinInterface'. Then click 'bin' > 'Debug'. Double click 'HashJoinInterface.exe' to launch the program. Alternatively, you may run the program in Visual Studios.

## Screenshots



**Caption:**  
Program at start.

Hash Join Algorithm Interface

R1 Instance:

R1-INSTANCE-1.csv

R2 Instance:

R2-INSTANCE-1.csv

Join

R1 Attribute:

a1

R2 Attribute:

b1

R1 Instance

**Caption:**

Selections made for drop down boxes.

Hash Join Algorithm Interface

R1 Instance

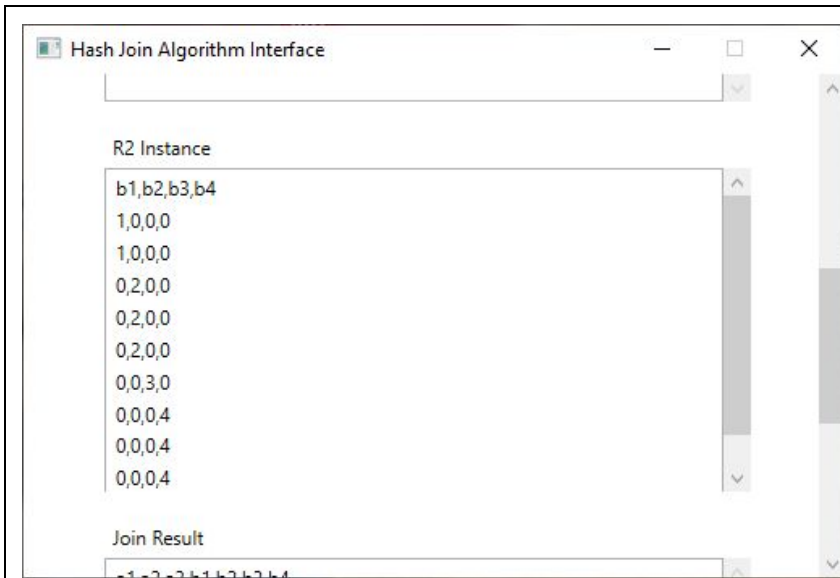
a1,a2,a3  
1,0,0  
2,0,0  
3,0,0  
4,0,0

R2 Instance

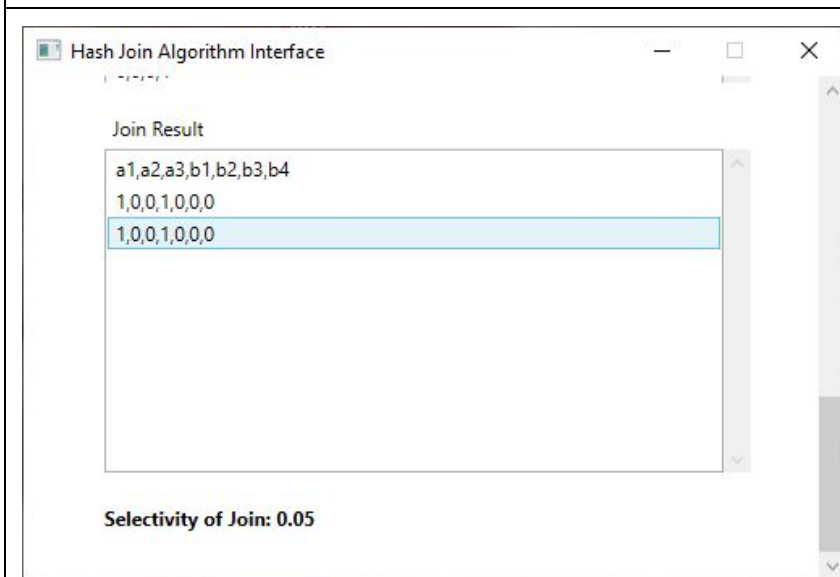
b1,b2,b3,b4

**Caption:**

Instance of R1 displayed.



**Caption:**  
Instance of R2 displayed.



**Caption:**  
Result and selectivity of join displayed.

## Sample Execution Outputs

Query	Input R1 Instance	Input R2 Instance	Output
$R1 \bowtie_{R1.a1=R2.b1} R2$	a1,a2,a3 1,0,0 2,0,0 3,0,0 4,0,0	b1,b2,b3,b4 1,0,0,0 1,0,0,0 0,2,0,0 0,2,0,0 0,2,0,0	a1,a2,a3,b1,b2,b3,b4 1,0,0,1,0,0,0 1,0,0,1,0,0,0  Selectivity=2/40=0.05

		0,0,3,0 0,0,0,4 0,0,0,4 0,0,0,4 0,0,0,4	
$R1 \bowtie_{R1.a1=R2.b2} R2$	a1,a2,a3 1,0,0 2,0,0 3,0,0 4,0,0	b1,b2,b3,b4 1,0,0,0 1,0,0,0 0,2,0,0 0,2,0,0 0,2,0,0 0,0,3,0 0,0,0,4 0,0,0,4 0,0,0,4 0,0,0,4	a1,a2,a3,b1,b2,b3,b4 2,0,0,0,2,0,0 2,0,0,0,2,0,0 2,0,0,0,2,0,0  Selectivity=3/40=0.075
$R1 \bowtie_{R1.a1=R2.b3} R2$	a1,a2,a3 1,0,0 2,0,0 3,0,0 4,0,0	b1,b2,b3,b4 1,0,0,0 1,0,0,0 0,2,0,0 0,2,0,0 0,2,0,0 0,0,3,0 0,0,0,4 0,0,0,4 0,0,0,4 0,0,0,4	a1,a2,a3,b1,b2,b3,b4 3,0,0,0,0,3,0  Selectivity=1/40=0.025
$R1 \bowtie_{R1.a1=R2.b4} R2$	a1,a2,a3 1,0,0 2,0,0 3,0,0 4,0,0	b1,b2,b3,b4 1,0,0,0 1,0,0,0 0,2,0,0 0,2,0,0 0,2,0,0 0,0,3,0 0,0,0,4 0,0,0,4 0,0,0,4 0,0,0,4	a1,a2,a3,b1,b2,b3,b4 4,0,0,0,0,0,4 4,0,0,0,0,0,4 4,0,0,0,0,0,4 4,0,0,0,0,0,4  Selectivity=4/40=0.1
$R1 \bowtie_{R1.a2=R2.b1} R2$	a1,a2,a3 0,5,0 0,6,0	b1,b2,b3,b4 5,0,0,0 5,0,0,0	a1,a2,a3,b1,b2,b3,b4 0,5,0,5,0,0,0 0,5,0,5,0,0,0

	0,7,0 0,8,0	0,6,0,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,0,8 0,0,0,8 0,0,0,8	Selectivity=2/52=0.0384
R1 $\bowtie$ <sub>R1.a2=R2.b2</sub> R2	a1,a2,a3 0,5,0 0,6,0 0,7,0 0,8,0	b1,b2,b3,b4 5,0,0,0 5,0,0,0 0,6,0,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,0,8 0,0,0,8 0,0,0,8	a1,a2,a3,b1,b2,b3,b4 0,6,0,0,6,0,0  Selectivity=1/52=0.0192
R1 $\bowtie$ <sub>R1.a2=R2.b3</sub> R2	a1,a2,a3 0,5,0 0,6,0 0,7,0 0,8,0	b1,b2,b3,b4 5,0,0,0 5,0,0,0 0,6,0,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,0,8 0,0,0,8 0,0,0,8	a1,a2,a3,b1,b2,b3,b4 0,7,0,0,0,7,0 0,7,0,0,0,7,0 0,7,0,0,0,7,0 0,7,0,0,0,7,0 0,7,0,0,0,7,0 0,7,0,0,0,7,0 0,7,0,0,0,7,0 0,7,0,0,0,7,0  Selectivity=7/52=0.134
R1 $\bowtie$ <sub>R1.a2=R2.b4</sub> R2	a1,a2,a3 0,5,0 0,6,0	b1,b2,b3,b4 5,0,0,0 5,0,0,0	a1,a2,a3,b1,b2,b3,b4 0,8,0,0,0,0,8 0,8,0,0,0,0,8



	0,7,0 0,8,0	0,6,0,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,7,0 0,0,0,8 0,0,0,8 0,0,0,8	0,8,0,0,0,0,8  Selectivity=3/52≈0.0576
R1 $\bowtie_{R1.a3=R2.b1}$ R2	a1,a2,a3 0,0,9 0,0,10 0,0,11 0,0,12	b1,b2,b3,b4 9,0,0,0 0,10,0,0 0,10,0,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,0,12 0,0,0,12	a1,a2,a3,b1,b2,b3,b4 0,0,9,9,0,0,0  Selectivity=1/64≈0.0156
R1 $\bowtie_{R1.a3=R2.b2}$ R2	a1,a2,a3 0,0,9 0,0,10 0,0,11 0,0,12	b1,b2,b3,b4 9,0,0,0 0,10,0,0 0,10,0,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0	a1,a2,a3,b1,b2,b3,b4 0,0,10,0,10,0,0 0,0,10,0,10,0,0  Selectivity=2/64=0.03125

		0,0,0,12 0,0,0,12	
R1 ⋈ <sub>R1.a3=R2.b3</sub> R2	a1,a2,a3 0,0,9 0,0,10 0,0,11 0,0,12	b1,b2,b3,b4 9,0,0,0 0,10,0,0 0,10,0,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,0,12 0,0,0,12	a1,a2,a3,b1,b2,b3,b4 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0 0,0,11,0,0,11,0  Selectivity=11/64≈0.171
R1 ⋈ <sub>R1.a3=R2.b4</sub> R2	a1,a2,a3 0,0,9 0,0,10 0,0,11 0,0,12	b1,b2,b3,b4 9,0,0,0 0,10,0,0 0,10,0,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,11,0 0,0,0,12 0,0,0,12	a1,a2,a3,b1,b2,b3,b4 0,0,12,0,0,0,12 0,0,12,0,0,0,12  Selectivity=2/64=0.03125
R1 ⋈ <sub>R1.a1=R2.b1</sub> R2	a1,a2,a3 1,1,1 1,1,1 1,1,1	b1,b2,b3,b4 2,2,2,2 2,2,2,2 2,2,2,2 2,2,2,2	a1,a2,a3,b1,b2,b3,b4  Selectivity=0

<div>R1</div> <div>⊗ R1.a1=R2.b1</div> <div>R2</div>	<div>a1,a2,a3</div> <div>1,0,0</div> <div>1,0,0</div>	<div>b1,b2,b3,b4</div> <div>1,0,0,0</div> <div>1,0,0,0</div>	<div>a1,a2,a3,b1,b2,b3,b4</div> <div>1,0,0,1,0,0,0</div> <div>1,0,0,1,0,0,0</div> <div>1,0,0,1,0,0,0</div> <div>1,0,0,1,0,0,0</div> <div>Selectivity=4/4=1</div>
--	---	--	--