Part a.

Description:

Mapper:

For each entry, set target as the key, and weight as the value.

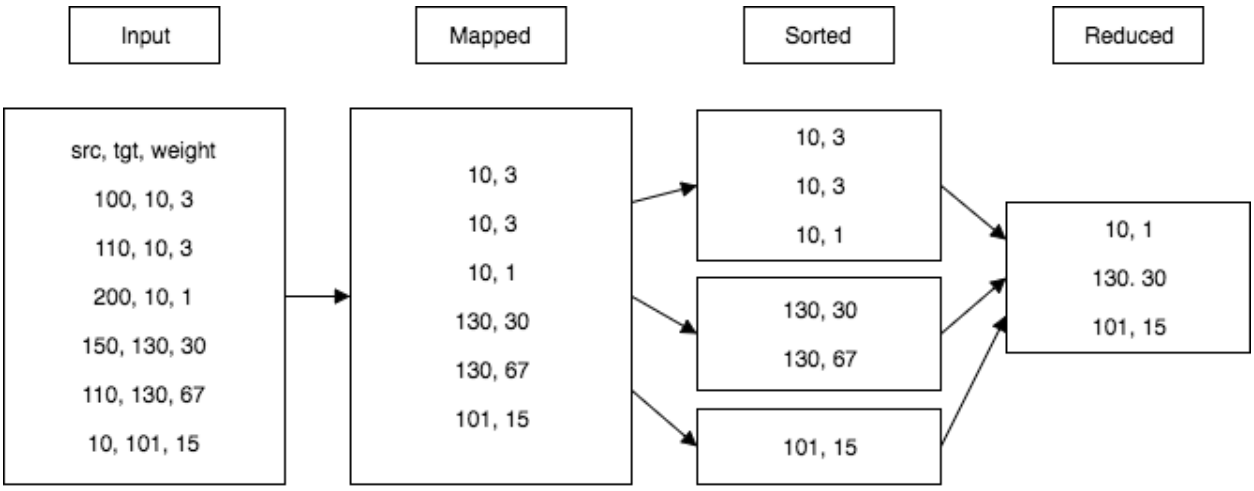
E.g.: [100, 10, 3] -> [10: 3]

Reducer:

For entries with the same key, set the minimum weight as the value.

E.g.: [130: 30], [130, 67] -> [130: 67]

Data Tracing:



Note: Splitting before map phase is omitted in this chart, because it does not affect my algorithm.

Part b.

Pseudocode:

```
class MyMapper
        function map (value)
                // the value here is a record entry of Student or Department.
                String record = value.toString();
                String[] attributes = record.split(“,”);
                if (attributes[0] == “Student”):
                        context.write(attributes[2], “student, ” + attributes[1]);
                if (attributes[0] == “Department”):
                        context.write(attributs[1], “department, ” + attributes[2]);
class MyReducer
        function reduce(key, values)
        // the key is the department number, which is already sorted by the MapReduce
        //framework
        String name = “”;
        String dept = “”;
        for (record : values):
                String[] attributes = record.split(“,”);
```

```
            if (attributes[0] == “department”):
                  dept = attributes[1];
                  break;
      for (record : values):
            String[] attributes = record.split(“,”);
            if (attributes[0] == “student”):
                  name = attributes[1];
                  context.write(key, name + “, ” + dept);
```

Description:

Mapper:

1. Take in a record entry as input. The record entry could be Student type of Department type.
2. Judge the type of the record.
3. Attach a tag representing the corresponding type of the record.
4. set the key as the department number, and the value as the student name or department name plus its corresponding type tag.

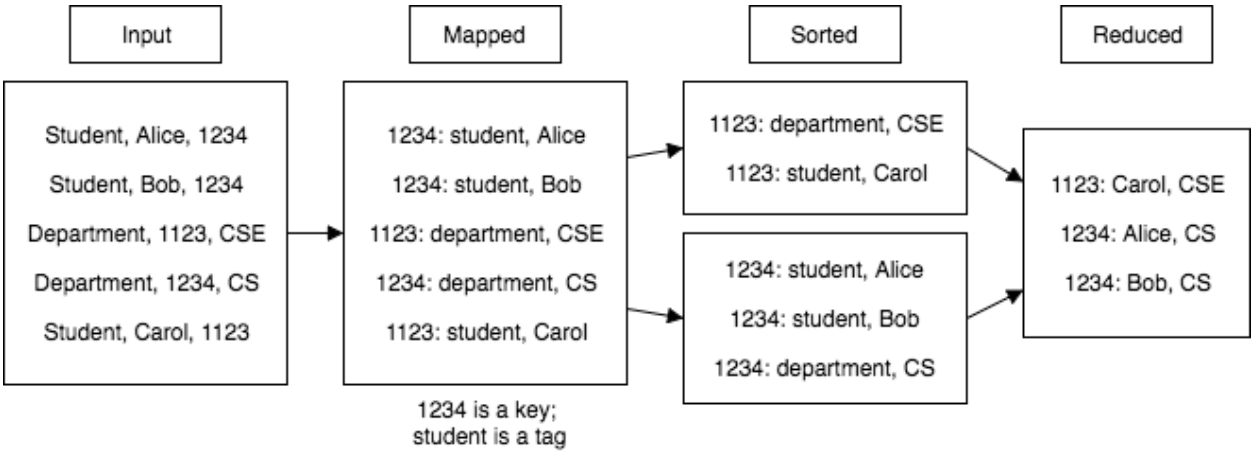
E.g.: [Student, Alice, 1234] -> [1234: student, Alice]

Reducer:

0. At this point, the key-value pairs would have been sorted by the key which is the department number.
1. For the values with the same key – department number, determine whether each value a student name or department name.
2. Get the department name first by finding the first department record, and set it as the department variable.
3. For each student record, set the student name as the name variable, and output the key and the name plus the department as required.

E.g.: [1234: student, Alice], [1234: student, Bob], [1234: department, CS] -> dept = Cs -> [1234: Alice, CS], [1234: Bob, CS]

Data Tracing:



Note: Splitting before map phase is omitted in this chart, because it does not affect my algorithm.