Pseudocode:

```
student_file <- student.txt
company_file <- company.txt
vector[101][101][1] <- a zero based 3 dimension vector
struct student <- studentID conduct_grade semester_grade
struct company <- companyID conduct_grade semester_grade salary
struct output <- studentID, companyID

sort(company_file) <- ID from small to big
sort(company_file) <- salary from high to low
g1 <- student.conduct_grade
g2 <- student.semester_grade
while (student_file not empty) do
    read line of student_file
    if(vector[g1][g2] is 0)
        vector[g1][g2][0] <- student.ID
    else
        create vector[g1][g2][n+1] //n is the number of student already in this grade
        vector[g1][g2][n+1] <- student.ID
    end if
end while

while (company_file not empty) do
    read line of company _file
    for i (100 to company. conduct_grade) do
        for j (100 to company.semester_grade) do
            output.studentID <- vector[g1][g2][n]
            output.compantID <- companyID
            vector[g1][g2][n] <- 0
        end for
    end for
end while
sort(output_file) <- ID from small to big
output_file <- output
output.txt <- output_file
```

Time complexity analysis:

Assume that there are n student and m company

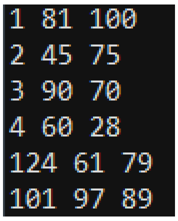
```
O(n)      student_file <- student.txt
O(m)      company_file <- company.txt
vector[101][101][1] <- a zero based 3 dimension vector
struct student <- studentID conduct_grade semester_grade
struct company <- companyID conduct_grade semester_grade salary
struct output <- studentID, companyID

O(mlogm)    sort(company_file) <- ID from small to big
O(mlogm)    sort(company_file) <- salary from high to low
g1 <- student.conduct_grade
g2 <- student.semester_grade
O(n)      while (student_file not empty) do
            read line of student_file
            if(vector[g1][g2] is 0)
                vector[g1][g2][0] <- student.ID
            else
                create vector[g1][g2][n+1]
                vector[g1][g2][n+1] <- student.ID
            end if
          end while
O(m*1*1)        while (company_file not empty) do
                    read line of company _file
                    for i (100 to company. conduct_grade) do
                        for j (100 to company.semester_grade) do
                            output.studentID <- vector[g1][g2][n]
                            output.compantID <- companyID
                            vector[g1][g2][n] <- 0
                        end for
                    end for
                end while
O(nlogn)    sort(output_file) <- ID from small to big
output_file <- output
output.txt <- output_file
```

2*O(n) + 2*O(m) + 2*O(mlogm) + O(nlogn) = ***O(n*logn)***

Experimental results:

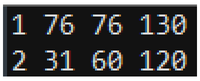
Example:

Student:



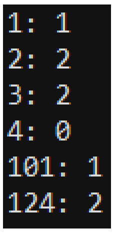
```
1 81 100
2 45 75
3 90 70
4 60 28
124 61 79
101 97 89
```

Company:



```
1 76 76 130
2 31 60 120
```

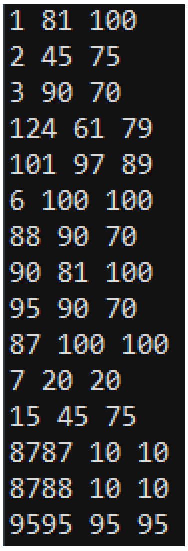
Output:



```
1: 1
2: 2
3: 2
4: 0
101: 1
124: 2
```

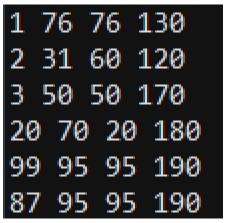
My own data:

Student:



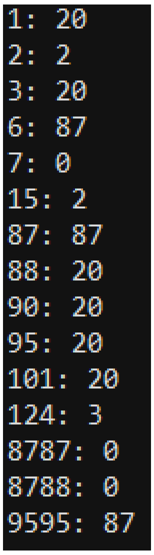
```
1 81 100
2 45 75
3 90 70
124 61 79
101 97 89
6 100 100
88 90 70
90 81 100
95 90 70
87 100 100
7 20 20
15 45 75
8787 10 10
8788 10 10
9595 95 95
```

Company :



```
1 76 76 130
2 31 60 120
3 50 50 170
20 70 20 180
99 95 95 190
87 95 95 190
```

Output:



```
1: 20
2: 2
3: 20
6: 87
7: 0
15: 2
87: 87
88: 20
90: 20
95: 20
101: 20
124: 3
8787: 0
8788: 0
9595: 87
```