Simon Anguish
Homework 4

1)

```
Sort jobs by $\frac{w_j}{t_j}$ descending;
Elapsed time = 0
Total weight = 0;
Jobs Scheduled = [];

For job in jobs {
        Elapsed time += job.time;
        Total weight = Elapsed time * job.weight;
        Jobs Scheduled[] = job;
}
```

Proving this with the exchange arguement involves taking a look at the sorted jobs. Assuming the jobs are sorted, such that job i has a lower weighted ration than job j, switching job i with job j would provide a different sum and thus different outcome. If the sum of the two jobs J' after the swap is higher than the sum of the two jobs J before the swap, then swapping i and j would prove to be not more optimal. If the swap does make the process more optimal, by performing these swaps, as long as the swap makes the processes more optimal, we may eventually be lead to an optimal sollution.

2)

Optimality for a classroom scheduling algorithm is bounded by a minimum number of classrooms needed as the maximum number of classes that occur at the same time. With the earliest-finish-time-first algorithm, the there are two possiblities when adding a new class to a classroom: If the start time is before the last end time of all set classrooms, we need to add a new classroom. If the start time is after the last end time of a set classroom, we can add that class to a pre-existing classroom. This will ensure that there will never be more classrooms in use than the maximum number of classes scheduled.