Okay, so solving successfully finished, and we've got the results.

If you had any trouble solving this problem, then you should go into Options in the Solver and select that you want to edit the length of solution time and increase it past 100 seconds.

The LibreOffice Solver is not incredibly powerful, so it may need more time to find the solution.

We do recommend that if you have access to Excel, that you use Excel instead for integer optimization problems, because Excel is a little bit better with integer optimization.

However, you probably got the solution just fine.

So you'll want to go ahead and click Keep Result.

So now, we can take a look at our solution.

So we've got the objective of 4.46.

This is not particularly interpretable, until you remember that what our objective is, is the total of all departments percent of target allocation hours reached.

So the best possible number that we could get here would be 5, if every department had 100% of their target allocation hours reached.

So we've got 4.46.

So let's go up and take a look at the solution.

So we're giving ophthalmology a total of four operating rooms per week, which is 32 weekly hours, or 81% of their target.

We're giving gynecology 14, which is 112 weekly hours, 95% of their target.

We're giving oral surgery two, and as we see, the constraints restricted us so that we only gave oral surgery operating rooms on Tuesdays and Thursdays.

This is 80% of the target.

We gave otolaryngology three operating rooms and general surgery 23 operating rooms, getting 91% and 97% of their target allocation hours.

So suppose that you were the operating room manager, who knows integer optimization.

In practice, what would happen is that you would go off and you would design what we just did here in the recitation.

You would have this optimization model for operating room scheduling in the hospital.

You'd solve it.

You'd find the optimal solution.

And you'd come back and present it to the surgical departments.

And they'll look at the schedule.

And all of a sudden, they'll tell you it doesn't work for them.

And you'll say, why not?

I incorporated everything that you asked for.

And it's true.

You did.

But oftentimes, there are actually other constraints that they forgot to tell you about.

So what would you do in that case?

Well, you'd listen to their additional constraints and go back and add them to the model.

So that's exactly what we'll do here.

Suppose the general surgery department took a look at the schedule and said, "Oh, great, we have 23 operating rooms per week.

That's exactly two less than actually our maximum.

It's right in between our minimum and our maximum.

And we're getting 97% of our weekly target.

We think that's wonderful.

However, it's a little bit uneven-- six on Wednesday, three on Thursday.

Can we just have at least four per day?" We can do that.

So let's go up to the minimum number of operating rooms per day.

And for general surgery, let's change this from zero to four.

This will help them balance throughout the week.

And potentially, ophthalmology will look at their schedule, and they'll think the same thing.

They'll think, "Well, we actually would prefer to balance it throughout the week.

We want at most one operating room per day.

We don't want to have two on Monday.

Let's put one of those on Thursday or Friday." So that's OK.

We can do that too.

We just go to the maximum number of operating rooms per day.

And for ophthalmology, we change it to one.

Let's re-solve the model and see what we get.

Again, we select the objective.

Go to Tools, pull down the Solver.

It should have everything already loaded up, and we just get to hit Solve.

So that was quick.

Solving successfully finished.

And it looks like we actually have exactly the same results as we had before.

Let's hit Keep Result and take a closer look at this.

So our objective function stayed exactly the same.

But we were able to incorporate the new constraints, the new requirement that the general surgery department have at least four operating rooms per day, and that the ophthalmology department have at most one operating room per day.

This means that within the solution we had before, there's actually a little bit of wiggle room.

We could move some things around without changing the optimality of the solution, that is, without changing the percent of target that each department received.

And as you see, this is a little more balanced for the ophthalmology department and the general surgery department.

So great-- so maybe seeing that this was available, then you'd go back, you'd present it, and general surgery would say, well, maybe could we have at least five per day?

So if you went up and changed this, made it five all the way across instead of four and you solved it again, what would happen?

You probably have an idea already.

And this is exactly what you were thinking.

No solution was found.

The model is infeasible.

Check limiting conditions.

So why is the model infeasible?

Well, let's take a look.

Scroll back over.

The model is infeasible because, let's see-- so it's not because of the minimum and maximum weekly department requirements, because general surgery said that they were willing to have up to 25 operating rooms per week.

However, if you look at the weekly targets, 25 times 8 is 200.

So if we were to actually assign general surgery five operating rooms per day, we would be assigning them 200 hours per week, which exceeds their weekly target.

And our optimization model says we cannot exceed the weekly targets.

So just right there, we can tell that it's infeasible because of this constraint.

So let's change this back to four.

So suppose you've come up with a solution, you've gone back to the surgical departments, you've presented your solution, they've maybe requested additional changes, and we've made those changes here.

So does this mean that your job is done?

Well, no, not really.

Of course, you could go back to the surgical departments and tell them that this is the schedule they're going to have every week.

But I bet if did that, you'd hear some complaints.

Let's go into our solution and take a closer look to see why.

So for example, let's look at the ophthalmology department.

In our solution, we're assigning them four operating rooms per week.

That gives them 32 weekly hours, which is 81% of their target allocation hours.

If you assigned the schedule to ophthalmology every week, then they would never get more than 81% of their target allocation hours.

And over time, this would lead to a real shortage.

The reason why we never assign them more than four in our optimization model is because their weekly target number of hours was 39.4.

If we had assigned them five operating rooms, it would be giving them 40 hours per week, which exceeds their weekly target.

So the way our optimization model is set up, we will never give them more than four operating rooms per week.

But if you were the OR manager, you would realize that this is actually something that we should be a little more flexible about.

Perhaps you would consider giving them five operating rooms one week, or maybe five operating rooms every two weeks out of the month.

This is feasible because we have extra operating rooms on Thursday and Friday, as you can see down here.

Plus, it seems like it's a more efficient and fair solution.

Yes, although they do exceed their weekly target by 0.6 hours, it means that on the weeks when they only get to have 81% of their target allocation, they know that the next week, they'll be able to make up for it.

So it seems like a more fair solution, as well as a more efficient solution.

By combining the power of integer optimization with the understanding and flexibility of human judgment, you really get the best of both worlds.