

Chapter 11

Optimizing Decisions

Some problems have many options that meet the requirements and business rules. The challenge is to choose the best option. In this chapter, you use Excel to help make the best choice between acceptable options and to justify that choice.

Why Do I Need to Know This?

The skills and abilities in this chapter are beyond what most people know about using Excel. When you master them, you will be able to perform feats of analysis that are unusual and impressive. Instead of saying on your resume that you know Excel, mention these skills to raise eyebrows and distinguish yourself from the dozens of other applicants who are after the job you want.

1 Modeling and Optimization

Learning Objectives

1. Identify characteristics of a computer model. [11.1.1]
2. Describe the iterative problem solving technique and how it is automated in Excel. [11.1.2]
3. Identify the characteristics of a model that is suitable for finding optimum solutions to problems using Solver. [11.1.3]

A typical **what-if analysis** is done by trying different values as inputs to observe the effect on formulas that depend on the values. When the formulas are related to more than one input and to the results of other formulas, then the combination reacts to changes in a manner that is similar to a real situation. By trying different combinations of inputs, the user can learn how the system of formulas reacts to changes. Changes to some inputs might have little effect while small changes to other inputs could have a large effect. This type of examination is called **sensitivity analysis**. Designing a set of interactive formulas that simulate a real situation is called **computer modeling**. Computer models can be very complex or relatively simple.

Computer modeling

The test of the value of a computer model is whether or not it accurately predicts the behavior of the system it is intended to represent. For example, a homeowner decides to use Excel to create a spreadsheet that calculates the balance on a home mortgage loan to simulate the payments and balances that are calculated by the bank. She designs a spreadsheet to simulate the repayment of a loan. The interest due each month is calculated using $1/12$ of the annual interest rate and added to the balance from the previous month before the monthly payment is deducted. This process is repeated until the loan is paid off and the balance is zero. The computer calculates the remaining balance to fifteen digits. She uses the currency format and sets the program to round off the display to two places. She does not know that the bank from which she has the loan uses a slightly different practice. After the new balance is calculated each month, they round it off to the nearest cent and that value is used to calculate the interest for the next month. The small difference between the homeowner's model and the way the bank actually calculates payments is small but after hundreds of payments, the difference becomes more than one cent and the balance due on the loan appears to be slightly higher than the homeowner's model predicted, as shown in Figure 11.1.

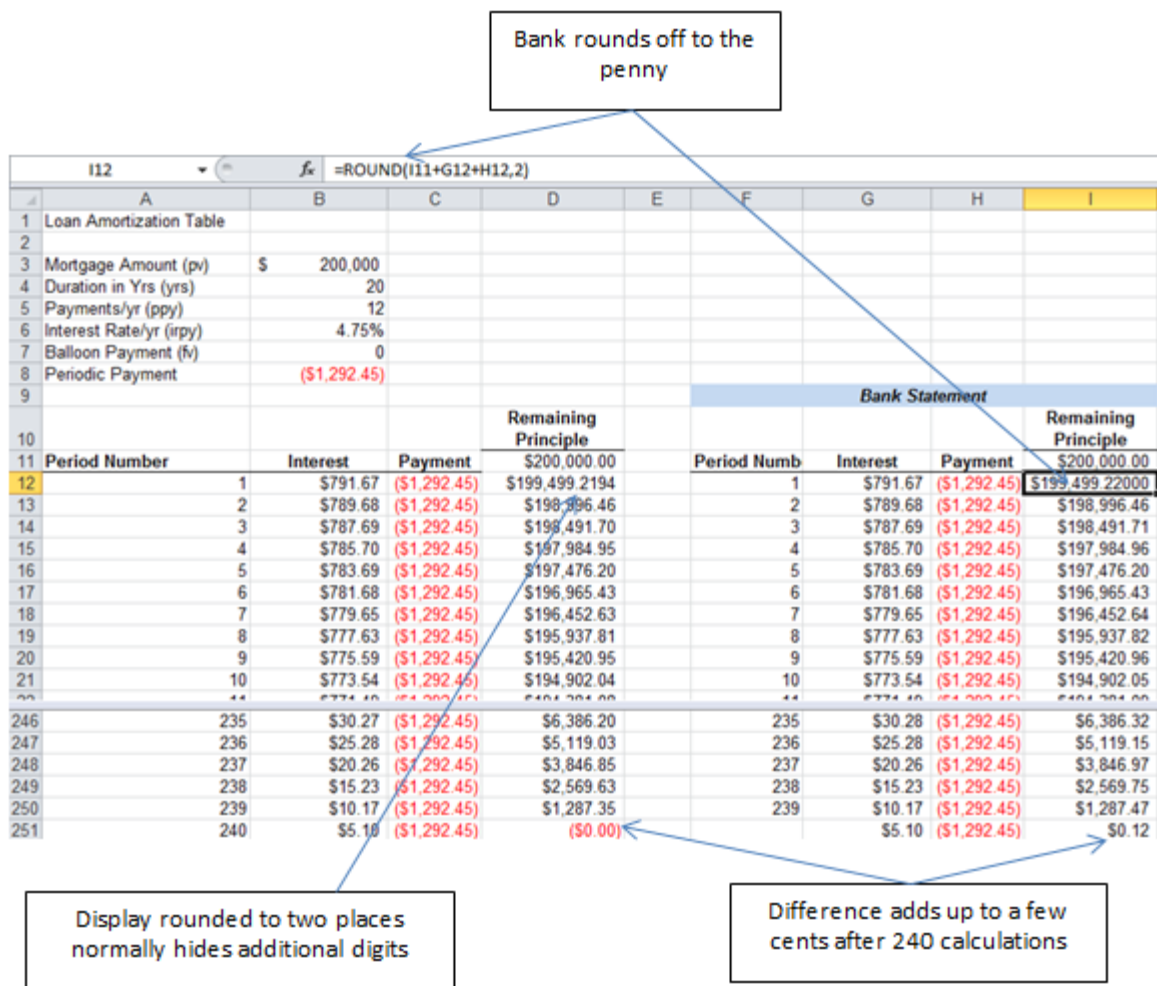


Figure 11.1. Rounding causes difference after many repetitions.

The homeowner's model is close enough to reality for most purposes but it illustrates an important point—rounding a display is not the same as rounding off the actual number. This small difference can become a big problem if business rules do not recognize that the numbers displayed might be slightly different than the underlying number. For example, if a city's billing program calculates the amount of a bill but does not round off the amount, the payment might not be exactly equal to the balance. If a business rule says to issue a dunning letter if the unpaid balance is greater than zero, the city can be embarrassed by sending out a dunning letter for zero cents when the balance was less than half a penny. The city of Attleboro, MA sent out a threat to a blind woman to put a lien on her property if she did not pay her outstanding utility bill for one cent (Kakley 2008) , as shown in Figure 11.2.

City wants her cent

BY MATT KAKLEY SUN CHRONICLE STAFF

Tuesday, November 18, 2008 2:40 AM EST



Eileen Wilbur, who has lived at 16 Glenn St. in South Attleboro since 1959, received a bill from the city Monday stating that if she does not pay an outstanding utility balance of one penny by Dec. 10 she will face penalties of up to \$48, along with a lien on her property. (Staff photo by Mark Stockwell)

Figure 11.2. Business rules must anticipate rounding errors.

In this instance, the computer did not have the world knowledge to know that it would cost the city more than the bill was worth to collect it, not to mention the embarrassment. As you recall from the previous chapter, world knowledge refers to background knowledge that is used to provide context for a piece of data. This story was circulated internationally and even showed up in the Tehran Times in Iran. (Tehran Times International Daily 2008)

An effective computer model should behave in a manner that is close enough to a real situation to achieve its intended purpose.

Iterative Problem Solving with Goal Seek

Recall that the Goal Seek function was used in a previous chapter to manipulate a model in which the cash flow and net present value were calculated for the purchase of an electric car, as shown in Figure 11.3. The Goal Seek function is limited to a single changing cell and the output must be a specific number.

	A	B	C	D	E	F	G	H
1	Electric Car							
2								
3	Price of electric car	\$ 31,000.0						
4	Price of comparable car	\$ 17,000.0						
5	Tax credit	\$ 7,500.00						
6	Premium paid for electric car	\$ (6,500.00)						
7	Miles driven per month	1,000						
8	Cost of gasoline per gallon	\$ 2.90						
9	Miles per gallon of comparable car	29						
10	Miles per kWh of battery	4.17						
11	Cost per kWh	\$ 0.08						
12	Savings per month on fuel	\$ 82.55						
13								
14	Year	2011	2012	2013	2014	2015	2016	2017
15	Cost and savings	\$ (6,500.00)	990.65	990.65	990.65	990.65	990.65	990.65
16								
17	Simple Payback	7						
18	ROI	-9%						
19	Discount Rate	6%						
20	NPV	(\$1,536.45)						

Figure 11.3. Financial model of the factors that influence the net present value.

The Goal Seek function is limited to a single changing cell and the output must be a specific number. To evaluate the effects of making changes to different inputs such as the tax credit, miles driven, or the cost of fuel, each of these must be done separately and stored as a scenario.

In some situations, the desired result is not known but it is known that it should be the minimum or maximum possible. Instead of a single changing cell, there might be several and each one could have a separate business rule that it has to satisfy. To create a model of these types of problems a more sophisticated function than Goal Seek is needed.

Optimizing Complex Models using Solver

If the value of the goal is not known but instead the goal is to find the minimum or maximum value, the more sophisticated **Solver** program can be used. It is found on the Data tab in the Analysis group. Like Goal Seek, Solver focuses its efforts on a single output cell and uses iterations of different values as inputs. The Solver program can vary more than one input at a time and place restrictions on the inputs that are called **constraints**. The user can request that the program find the minimum or maximum value without specifying what it is. For

example, a manufacturer of hardware supplies is evaluating a product that is a package containing a mixture of nails of different size. It is intended for apartment owners who just want a small quantity of nails available for small projects. The nails range in size from 1.5 inches up to 4 inches and a table is created that shows the values for size, cost, price, and profit for each nail, as shown in Figure 11.4.

	A	B	C	D	E	F	G	H
1	Package of Nails							
2								
3	Nail (in)	Weight (oz)	Cost (cents)	Price (cents)	Profit on each nail (cents)	Number	Total Weight (oz)	Profit (cents)
4	1.5	0.10	1.5	2	0.5		0	0
5	2.0	0.25	2.0	3	1.0		0	0
6	2.5	0.50	2.5	4	1.5		0	0
7	3.0	0.70	3.0	5	2.0		0	0
8	3.5	0.85	3.5	6	2.5		0	0
9	4.0	1.00	4.0	8	4.0		0	0
10					Totals	0	0	0

Figure 11.4. Model of a package of nails.

The marketing department has provided the following business rules:

1. There cannot be more than ten of any one type of nail
2. The total number of nails cannot exceed thirty
3. The total weight cannot exceed twenty ounces
4. Find the mix that produces the greatest profit

The business rules from the marketing department are translated into rules that are a better match to logical operators:

1. Number ≤ 10
2. Sum of numbers ≤ 30
3. Sum of total weight ≤ 20
4. Maximize the total profit

The objective of the project is to find the mix of nails that meets the requirements and also maximizes the total profit. The solver program is started and the problem is described as shown in Figure 11.5.

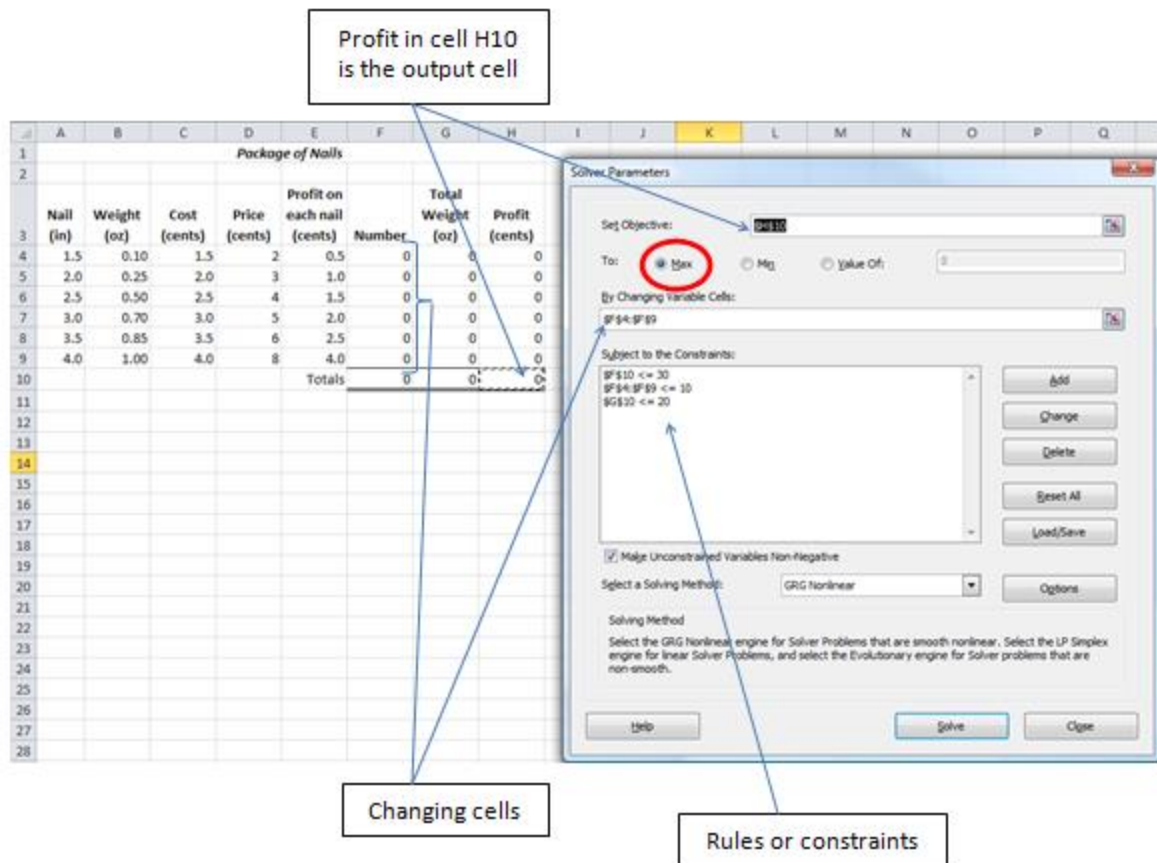


Figure 11.5. Solver uses multiple input cells to find minimum or maximum outputs.

The business rules are added to the dialog box as constraints. The marketing department assumed certain world knowledge that the computer does not have. When the program is run by clicking the Solve button, the results are unexpected, as shown in Figure 11.6.

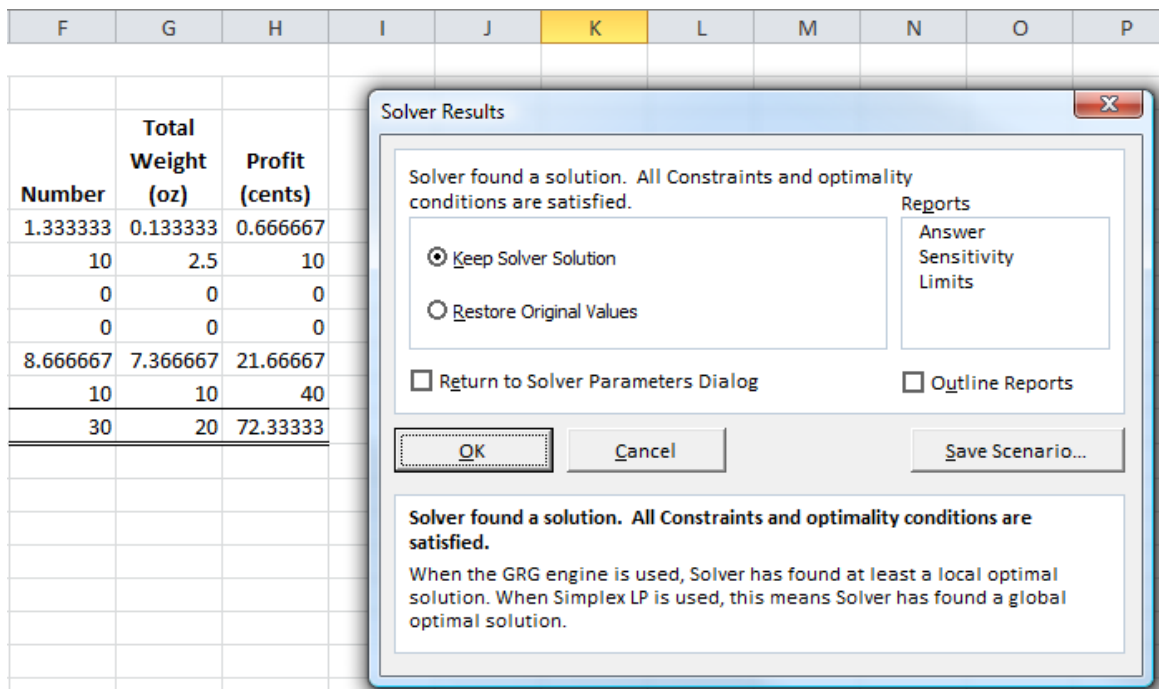


Figure 11.6. Program produces unexpected results.

The marketing department assumed that everyone knows that you do not sell fractions of nails.

Another constraint is necessary that specifies that the numbers of nails must be **integers**—whole numbers.

This time, the result is more useful, as shown in Figure 11.7.

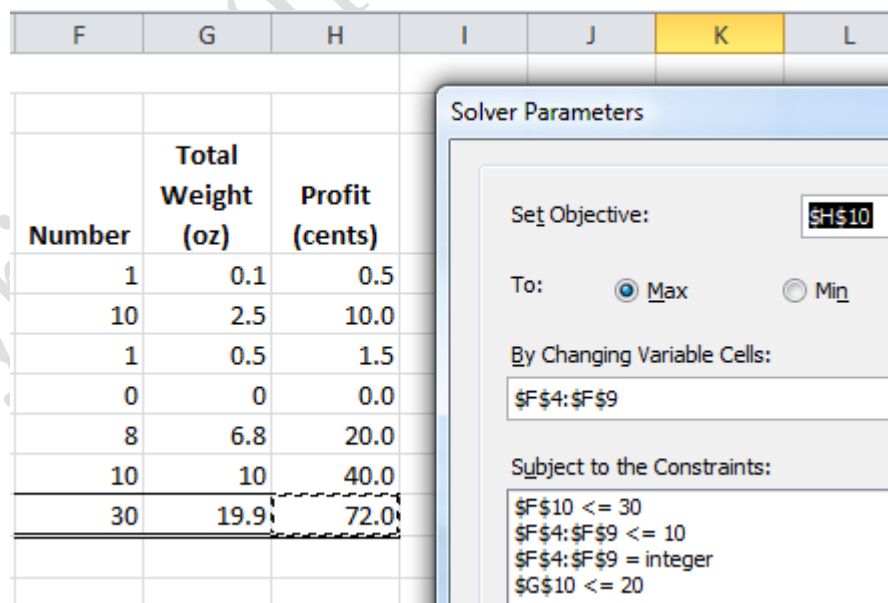


Figure 11.7. Program results are improved by requiring whole numbers of nails.

Even though this result satisfies the initial rules, the marketing department is not happy with the result. They would like to perform a sensitivity analysis to know what it would do to the profit of the package if there is a minimum for each type of nail of 1, 2, 3, 4 or 5 nails. An additional constraint is added that imposes a minimum of one nail for each type, as shown in Figure 11.8.

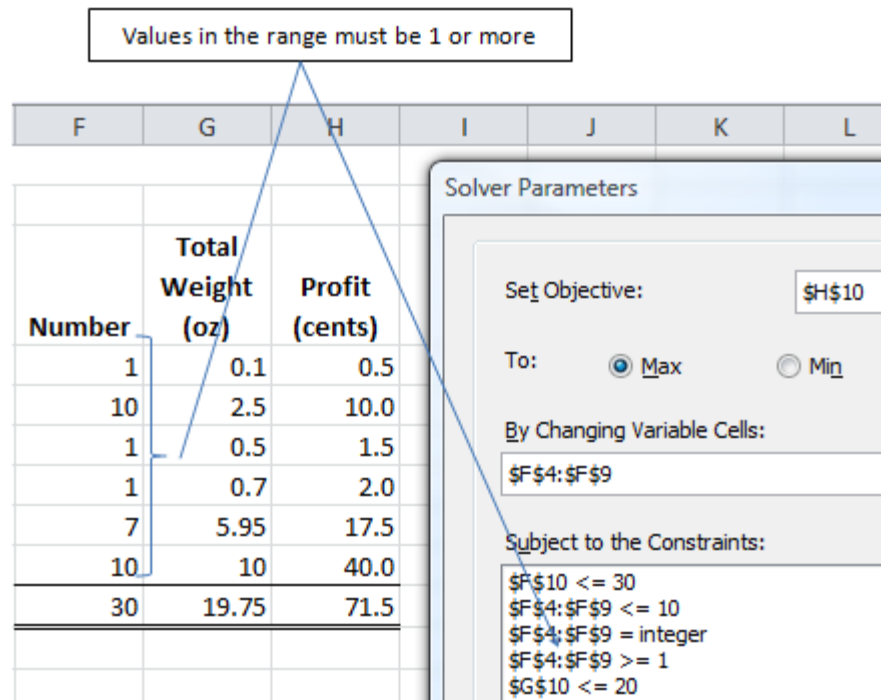


Figure 11.8. Require at least one nail of each type.

The result is stored as a scenario named Min1. The constraint is edited and changed to require at least two nails. The result is saved and the process repeated. Solutions that are possible are called **feasible**. If the requirement is changed to a minimum of six nails for each type it conflicts with another constraint. It is impossible to require a minimum of six nails in each of the six types because the total would be more than thirty. When there is no feasible solution, the program displays an error message and allows the original values to be restored, as shown in Figure 11.9.

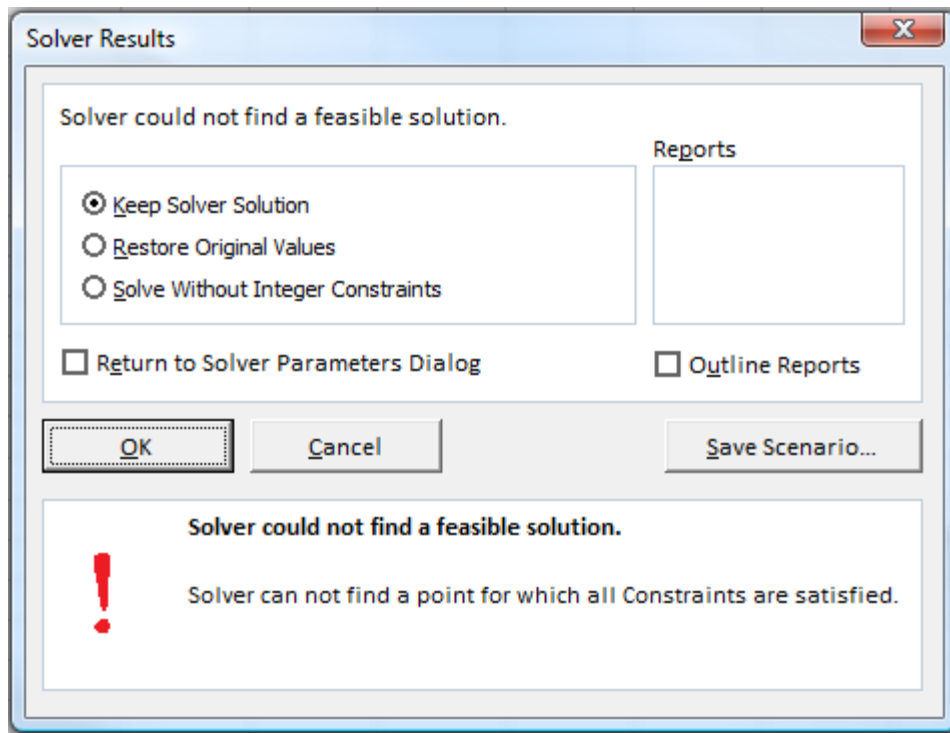


Figure 11.9. Warning for impossible combination of constraints.

Six feasible scenarios are recorded where the minimum value varies from zero to five from which a PivotTable is created and sorted by the profit, as shown in Figure 11.10.

	A	B
1	\$F\$4:\$F\$9 by (All)	
2		
3	Row Labels	\$H\$10
4	NoMin	72.0
5	Min1	71.5
6	Min2	71.0
7	Min3	70.5
8	Min4	70.0
9	Min5	57.5

Total profit drops dramatically from four to five nails as a minimum

Figure 11.10. Sensitivity analysis shows how profit varies with nail minimum.

Examination of the table shows that the total profit on the package falls about a half cent for each increase in the minimum but then there is a dramatic drop when the minimum is increased from four to five. The marketing department is happy to have this information. They know that a package with an equal

number of nails (six types with five each) would be much less profitable than any of the other options. They will combine their qualitative world knowledge of customer behavior and competing products with this quantitative analysis to make a decision.

Using Solver to Assign Contracts

Solver can be used to minimize costs by finding the best combination of competing options called the **assignment model** in this text. The assignment model makes use of the Excel SUMPRODUCT function. This function uses two ranges of equal size as its arguments. The function matches up the cells in the two ranges in pairs, multiplies each pair and then sums the result, as shown in Figure 11.11.

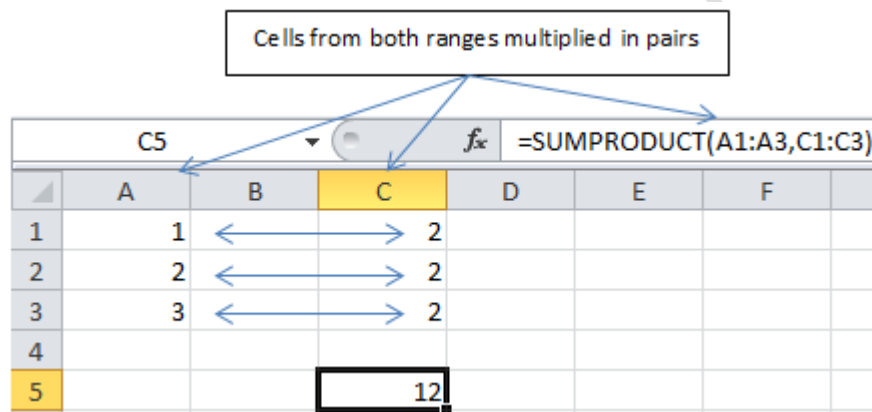


Figure 11.11. SUMPRODUCT function multiplies ranges pairwise and then sums.

This function can be used to award contracts. For example there are three contracts—A, B, or C—to be awarded and four companies have submitted bids. The SUMPRODUCT function is used to determine the total contract cost and SOLVER is used to find the combination of awards that has the least cost. Examine the table of bids as shown in Figure 11.12 and read the explanation below the figure.

E10		fx		=SUMPRODUCT(B5:B8,E5:E8)			
	A	B	C	D	E	F	G
1	Awarding Contracts						
2							
3		Bids on Contracts (\$1,000s)			Assignment Awarded		
4	Company	Site Preparation	Towers and Turbines	Electrical Systems	Site Preparation	Towers and Turbines	Electrical Systems
5	Prarie Wind	75	105	32			
6	Wind Towers	80	110	27			
7	Future Power	70	95	29			
8	Clean Wind	65	120	35			
9				Awarded	0	0	0
10				Winning Bid	0	0	0
11						Total	0

Figure 11.12. Bids for contracts are listed in a table.

The bids from each company for each contract are listed in columns B, C, and D. In cells E9:G9 a simple SUM function is used to add up the contracts that will be awarded. Only one bidder will be assigned to each contract. In cells E10:G10, the SUMPRODUCT function is used to calculate the cost of the winning bid for each contract. In cell G11 the sum function is used to sum the costs of the winning bids in row 10. Solver is set to minimize the overall cost in cell G11. Then in cells E5:G8 it assigns a 1 for the winning bid for each contract, and a 0 for the other bids. The SUMPRODUCT function multiplies the 1 or 0 by the bid amount for each contract, which is then totaled in cell G11. When Solver is run, the results are as shown in Figure 11.13.

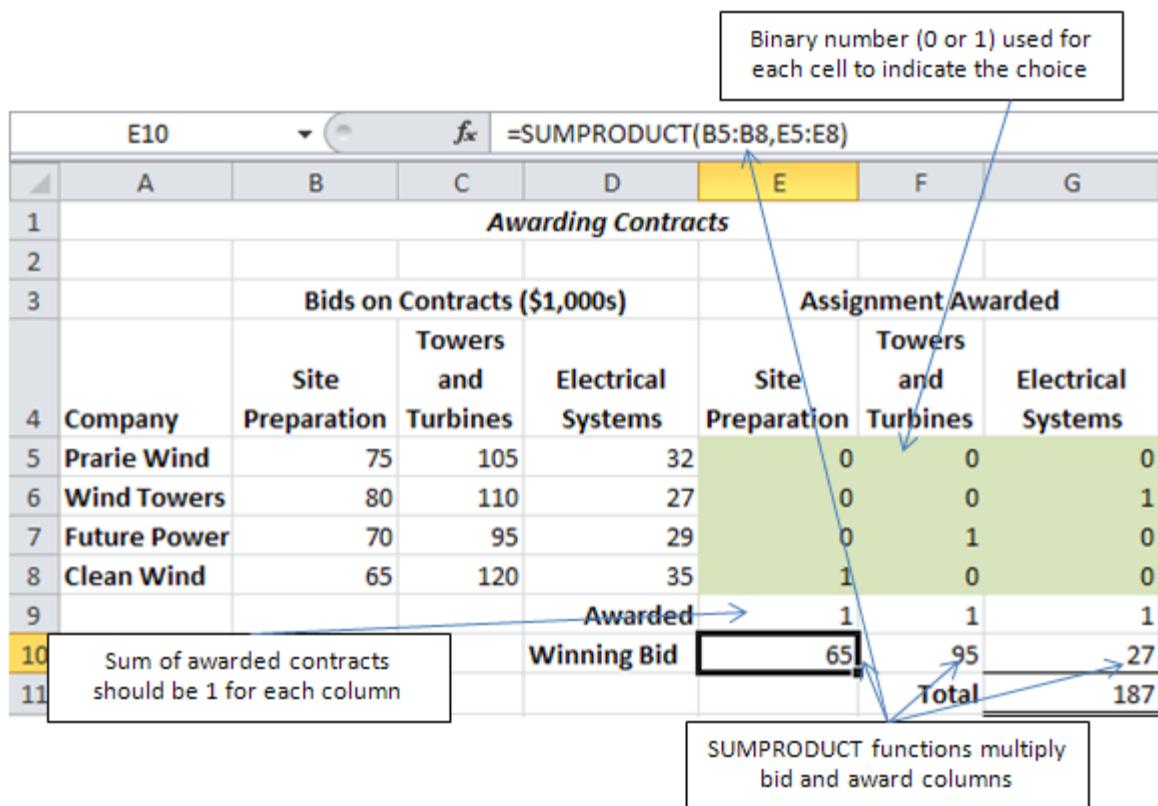


Figure 11.13. SUMPRODUCT function multiplies bids by the awards.

The Solver program seeks the minimum total cost in cell G11. The process is:

- Solver uses binary numbers (0 for No or 1 for Yes) to assign the contract in cells E5:G8
- The sum of the number of awards for each contract in row 9 is 1
- The SUMPRODUCT function calculates the cost of each bid that is awarded
- The lowest bid is at least 1 (to avoid the problem of rounded number that are not exactly zero)

The constraints used in Solver are shown in Figure 11.14.

	A	B	C	D	E	F	G	H	I	J
1	Awarding Contracts									
2										
3		Bids on Contracts (\$1,000s)			Assignment Awarded					
4	Company	Site Preparation	Towers and Turbines	Electrical Systems	Site Preparation	Towers and Turbines	Electrical Systems			
5	Prarie Wind	75	105	32	0	0	0			
6	Wind Towers	80	110	27	0	0	1			
7	Future Power	70	95	29	0	1	0			
8	Clean Wind	65	120	35	1	0	0			
9				Awarded	1	1	1			
10				Winning Bid	65	95	27			
11						Total	187			
12										

Solver Parameters

Set Objective:

To: ☐ Max ☒ Min

By Changing Variable Cells:

Subject to the Constraints:

\$E\$10:\$G\$10 >= 1

\$E\$5:\$G\$8 = binary

\$E\$9:\$G\$9 = 1

\$H\$5:\$H\$8 <= 1

Figure 11.14. Constraints for contract awards.

Using Solver to Assign Shifts

A similar method can be used to assign workers to particular shifts. In that model, the zeros and ones are typed in the worksheet to create a pattern of workdays and the Solver program assigns people to those shifts to meet minimum staffing needs. In this model, the 1s represent days worked and the computer chooses the numbers of workers assigned to each shift in column C, as shown in Figure 11.15.

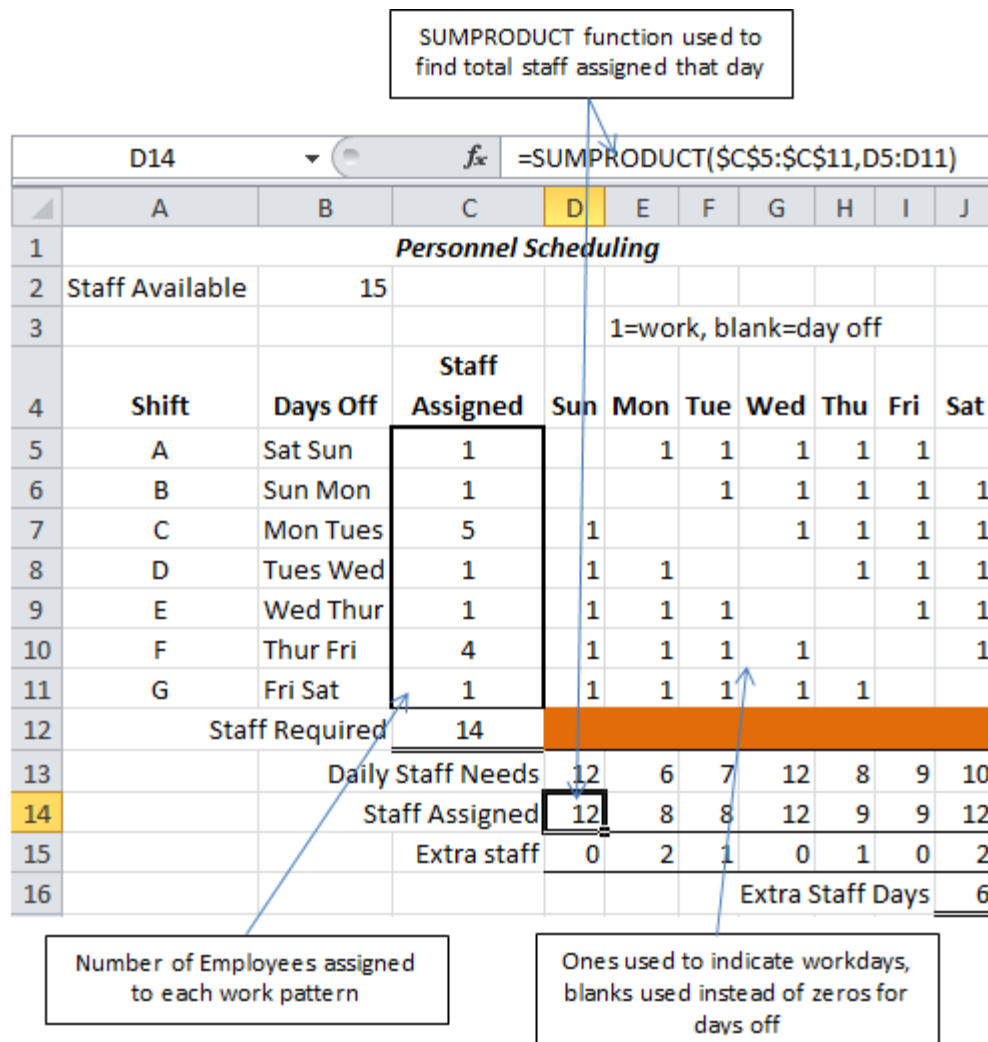


Figure 11.15. In column C, Solver chooses how many people to assign to each work pattern.

Key Takeaways

- A computer model is a set of interactive formulas that are designed to respond in a way that is similar to a real system. [11.1.1]
- An iterative problem solving is an automated trial-and-error method that uses feedback from the output value to choose the next input. Excel provides two functions, Goal Seek and Solver that use this method. Both functions focus on one output cell. Goal Seek is limited to finding an exact match by changing one input. Solver can find exact matches, minimums and maximums using several cells as inputs that vary within specified constraints. [11.1.2]
- To use the Solver function, a spreadsheet should be set up with clearly identified and separated inputs, each

in its own cell. Assumed values should be taken out of formulas if it is possible they might change. The cells should have descriptive names to make the dialog boxes and PivotTables easier to read. Colored backgrounds can be used to identify changing cells, the output cell and ranges used for constraints. [11.1.3]

2 Optimizing Decisions

Learning Objectives

1. Create a loan repayment schedule with an option for additional payments. Use Goal Seek to create early repayment scenarios. [11.2.1]
2. Use Goal Seek to find the Internal Rate of Return. [11.2.2]
3. Use Solver to find optimum solutions for assigning contracts or employees. [11.2.3]
4. Use dependency arrows to analyze formula relationships. [11.2.4]

Computer spreadsheets were invented to perform what-if analyses of financial projections where each year's values were calculated from the previous year's results. If a loan repayment schedule is designed with the appropriate conditional operators, it can be used to answer questions such as; "How much more would I have to pay each month to pay off my twenty-year loan in only fifteen years and how much interest would I save?" Recall that an amortization table shows the remaining balance on a loan each month after the interest has been added and the payment has been subtracted.

Create an Amortization Table

An amortization table shows the terms of the loan and includes a table listing the principal and interest payments and remaining balance after each payment for the length of the loan. To make the table useful for what-if analysis, the formulas are nested within IF statements that check to see if the loan is already paid off to prevent showing negative balances.

1. Navigate to the folder where files for this book are located. Open Ch11Optimize.xlsx and then save the file as Ch11OptimizeStudentName.

2. In the Ch11OptimizeStudentName workbook, click the Loan sheet tab. In the first few rows of this worksheet is the calculation for a monthly loan payment on a \$120,000 loan at 4.75% interest for 20 years. Click cell B7 and examine the PMT function displayed in the Formula Bar.
3. In cell A12, type **January 1, 2011**, and then on the Formula Bar, click the Enter button ☒.
4. Point to the fill handle in cell A12 and use the **right** mouse button to drag the fill handle down to cell A251. If you go too far, continue to hold down the right mouse button and point to the top of the worksheet until row 251 displays, and then release the mouse button. On the shortcut menu that displays as shown in Figure 11.16, use the left mouse button to click Fill Months. The final date in cell A251 is 1-Dec-30.

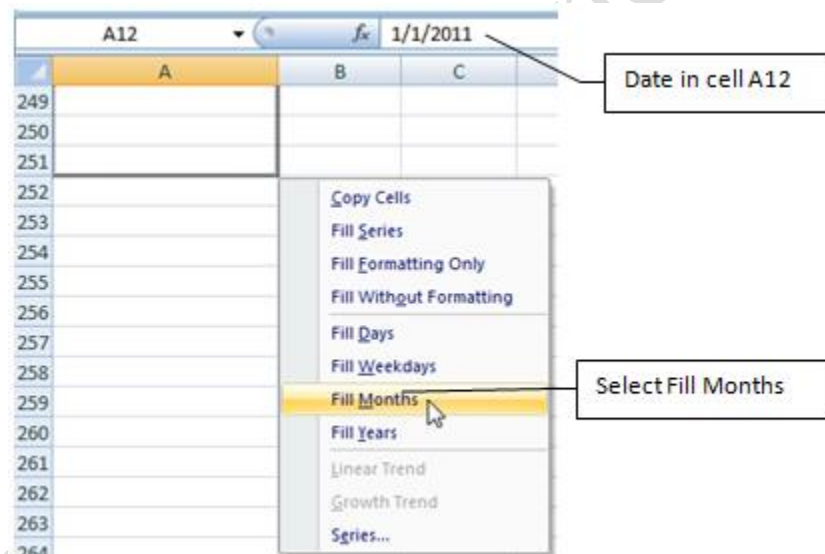


Figure 11.16. A shortcut menu displays options when filling with the right mouse button.

Tip: When you fill a range using the right mouse button a shortcut menu displays which gives you options to copy the cell, fill a series, fill formatting only, or fill without formatting. In the case of a date, if you used the left mouse button sequential dates would display, but sometimes you may want only weekdays, months, or years. The series created for the amortization table is the payment date, which is the same day of each month for twenty years. Using the right mouse button displays the shortcut menu which enables you to change the fill option to months.

5. Press Ctrl + Home to return to the top of the worksheet. In cell D11 type **=B3** and format the cell to the Accounting Number Format with two decimal places. The initial loan amount displays at the top of the

amortization schedule as the remaining balance. As payments are made this balance will decrease. Each payment consists of interest and payment toward the principal value of the loan (pv).

- Click cell B12 and display the IF Function Argument dialog box. In the Logical_test box type **D11>.01** In the Value_if_true box type **D11*B6/B5** and in the Value_if_False box type **0**. Compare your dialog box with Figure 11.17. The logical test is used to determine if there is a balance owed. It is better to test for a balance that is greater than a penny rather than a balance that is greater than zero because there might be rounding errors where the remaining balance is not exactly zero. If the balance is greater than zero, then the interest for one payment (B6/B5) is applied to the remaining balance. If not, then zero will display.

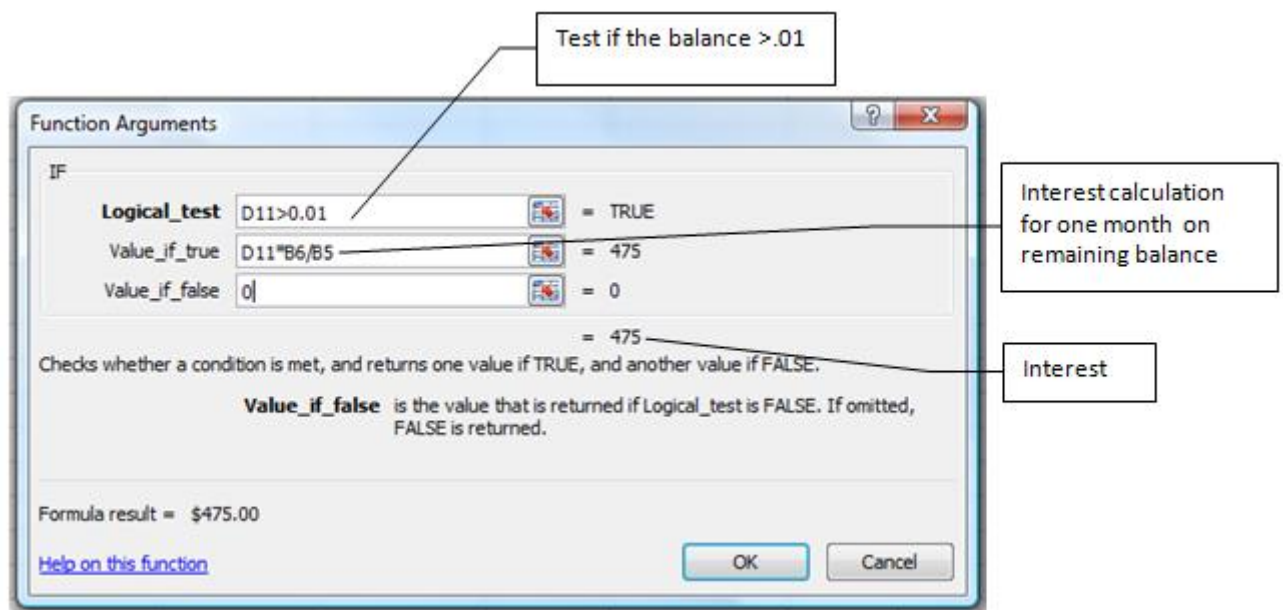


Figure 11.17. If there is a balance the interest on the remaining balance will display.

- In the Value_if_true box, select B6/B5 and press the F4 key to make this an absolute reference to the interest rate and the payments per year cells; then click OK. Because this formula will be filled down the column, the reference to the interest rate and payments per year must be absolute. The reference to the remaining balance is a relative cell reference because it will change as payments are made.

8. Click cell B9. Look at the formula and notice that this cell is a formula that adds an optional additional payment—cell B8—to the regular payment in cell B7.
9. In cell C12, type **=IF(D11>0.01,\$B\$9,0)** and then click the Enter button. This formula will display the payment from cell B9 when the remaining balance is greater than a penny, otherwise it will display zero.
10. In cell D12, type **=IF(D11>0.01,D11+B12+C12,0)** and compare your screen with Figure 11.18.

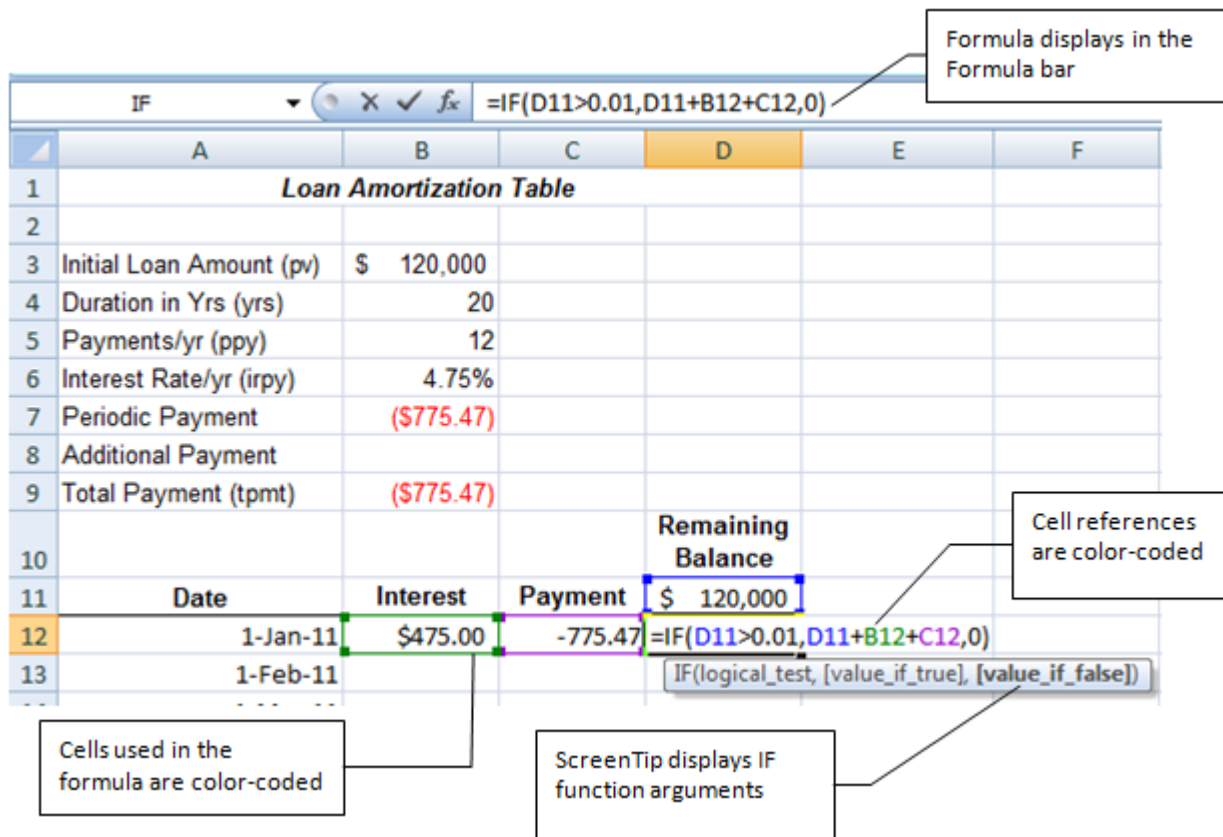


Figure 11.18. If there is a balance, then the remaining balance, interest, and loan payment are summed to create a new loan balance.

11. Press Enter to finish the formula. Select cells B12:D12 and format the cells with the Comma Style format.
12. With cells B12:D12 still selected, in the lower right corner of cell D12 double-click the fill handle to fill all three formulas down to row 251 where the final date for the loan displays.
13. Be sure the top of the worksheet displays on the screen, then click cell A12. On the View tab, in the

Window group, click the Split button. This splits the screen at the point of the active cell. A bar displays horizontally across the screen between row 11 and row 12 and two scroll bars display enabling you to scroll the top and bottom portions of the screen separately.

Tip: To remove the split you can click the Split button again, or double-click the bar on your screen. If the active cell had been in any column other than column A, then there would also be a vertical bar displayed along with two horizontal scroll bars. Then the screen would be split both horizontally and vertically.

14. On the bottom half of the screen, scroll down until row 71 displays (Dec 1, 2015). This is five years from beginning of the loan.
15. Select cells A71:D71. Right-click the selected cells, and on the Mini toolbar, click the Fill Color arrow. In the color palette, in the second row, third column click to apply the *Tan, Background 2, Darker 10%* fill color. When you point to each of the color chips a ScreenTip displays the name of the color as shown in Figure 11.19.

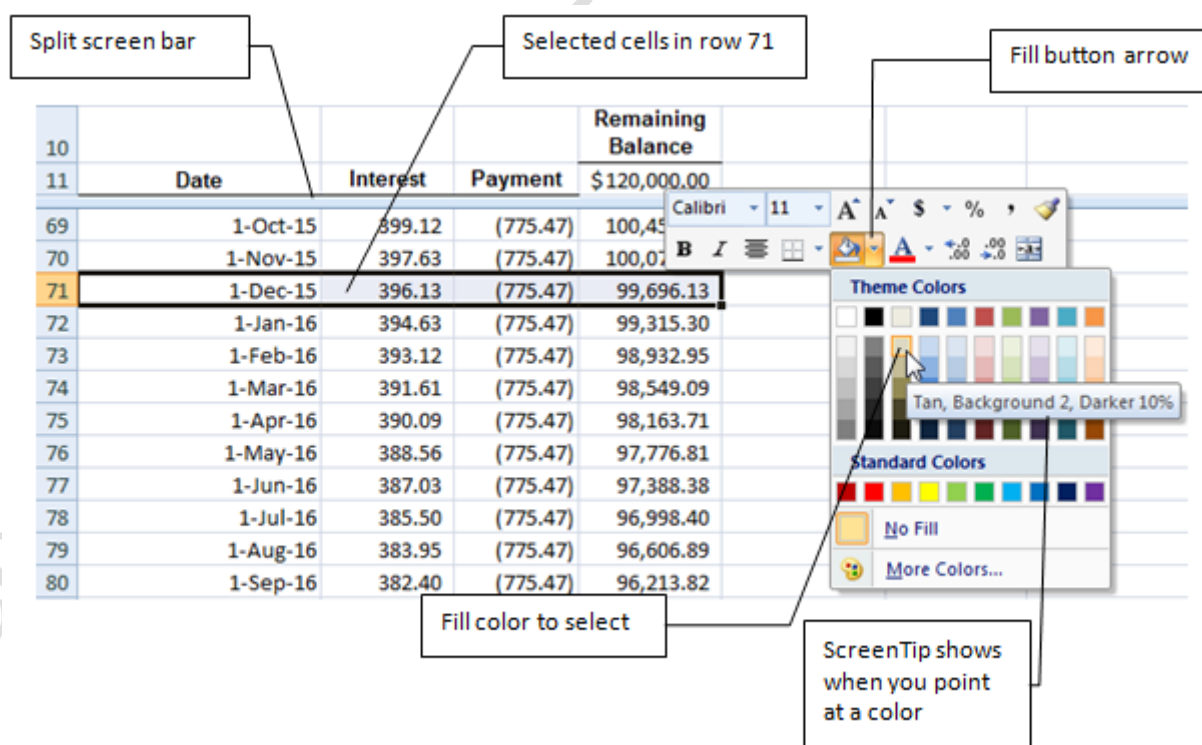


Figure 11.19. A fill color is applied to the values for a payoff in 5 years on 1-Dec-15.

16. Scroll down to display row 131 for 1-Dec-20. Select cells A131:D131 and apply the same fill color to these cells. Scroll to row 191(1-Dec-25) and apply the same fill color to cells. The rows for five, ten and fifteen years of payments are now shaded.
17. Scroll to the end of the amortization schedule. In B252, enter the SUM function to add cells B12:B251 to find a total of all the interest paid. Format the cell with the Accounting Number Format.
18. With cell B252 selected, in the Name Box type **Total_Interest** and then press Enter.
19. Scroll the bottom half of the window up to display row 209. In the top half of the window, click cell B8, type **-100** and press Enter. Recall that the payment column references cell B9, which is the sum of the regular payment and the additional payment in cell B8. The additional payment is entered as a negative number because it is considered cash paid out when you calculate cash flow. By paying an extra \$100 each month, the loan would be paid off by 1-Jun 27.

Tip: When you enter extra payment amounts, it is possible that the last payment could be larger than the balance from the previous month. This situation would result in a negative balance. It would be possible to write additional IF statements to cover this possibility but it would make the formulas more complex. The goal of creating a model is to be accurate enough to meet to objectives of the model—not to be perfect. You must be aware of the assumptions and limitations of computer models and watch for situations in which they differ from reality.

20. Scroll the bottom of the screen to display row 131 for a 10 year payoff. In cell B8 enter a negative amount to guess how much extra you would need to pay each month to pay off the loan in 10 years instead of 20. Change your guess three more times attempting to get closer to a payoff by December 1, 2020.

Use Goal Seek to Compare Alternatives and Save as Scenarios

By guessing you can get fairly close to the additional payment required to pay off the loan early, but the Goal Seek tool will find the exact amount that is necessary. If you want to compare the alternatives for paying off the loan early, you can save each of the Goal Seek results as a scenario by using the Scenario Manager.

1. In cell B8, delete any amount that displays. This returns the loan amortization table to the original 20 year loan term. Use the Name Box to name this cell **Additional_Payment**.
2. On the Data tab, in the Data Tools group, click the What-If Analysis button and then click Scenario Manager. The Scenario Manager enables to save several alternative what-if analysis solutions and create a report so that you can compare the results side by side.
3. In the Scenario Manager dialog box, click the Add button. In the Add Scenario dialog box, in the Scenario name box type **20-years** to name the first scenario. Confirm that B8 displays in the Changing cells box and then click OK. In the next dialog box, in the AdditionalPayment box, type **0** and then click OK. This preserves the original scenario for a 20-year loan. Close the Scenario Manager dialog box.
4. Scroll the bottom of the screen to display row 191. Click cell D191 to make it the active cell.
5. On the Data tab, in the Data Tools group, click the What-If Analysis button, and then click Goal Seek. Goal Seek is used to find a solution by changing one variable at a time. In this example, the goal is to pay off the loan after 15 years by changing the amount of the additional payment in cell B8.
6. In the Goal Seek dialog box, confirm that the Set cell box displays D191. In the *To value* box type **0**, in the *By changing cell* box type **B8**. Click OK. Goal Seek finds an answer that will pay off the loan by 1-Dec-25 by changing the additional payment to \$157.93. Compare your results with Figure 11.20.

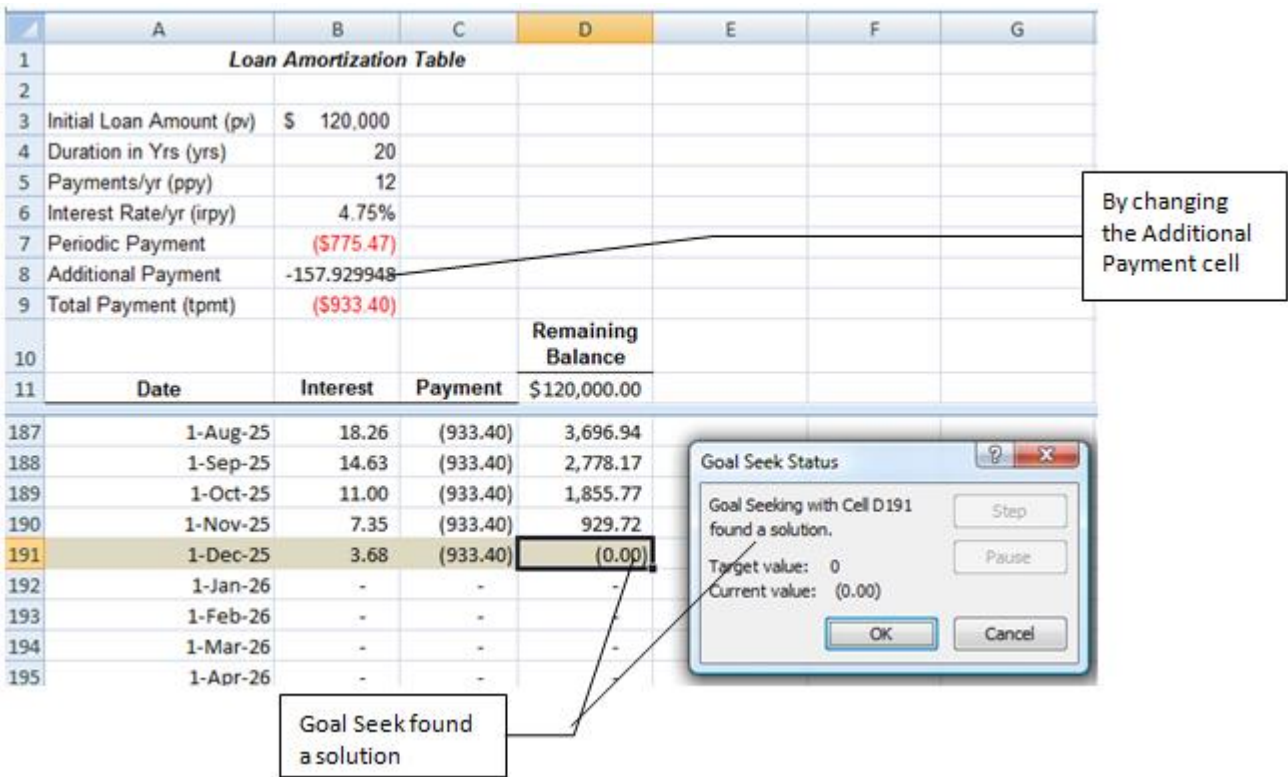


Figure 11.20. Goal Seek found a solution.

- Click OK in the Goal Seek Status dialog box. Copy the format from B7 to B8.
- On the Data tab, in the Data Tools group, click the What-If Analysis button and then click Scenario Manager. In the Scenario Manager dialog box click Add, and then type **15-years** to name this second scenario. Click OK twice and then close the Scenario Manager dialog box.
- Scroll the bottom of the window to display row 131. Click cell D131. Display the Goal Seek dialog box. Confirm that cell D131 displays as the Set cell. In the To value box type **0**, and in the By changing cell box type **B8**. Click OK two times.
- Display the Scenario Manager dialog box. Click Add and name this scenario **10-years**. Click OK twice, and then close the Scenario Manager dialog box.
- Scroll the bottom of the window to display row 71. Click cell D71. Use Goal Seek to find the extra

payment need to make the balance in cell D71 equal to zero.

12. Display the Scenario Manager dialog box and save this scenario as **05-years**. (The name 05 is used so it sorts properly in the Scenario Report). Click OK twice, but leave the Scenario Manager dialog box open. After all four scenarios have been created, the Scenario Manager lists the alternatives as shown in Figure 11.21.

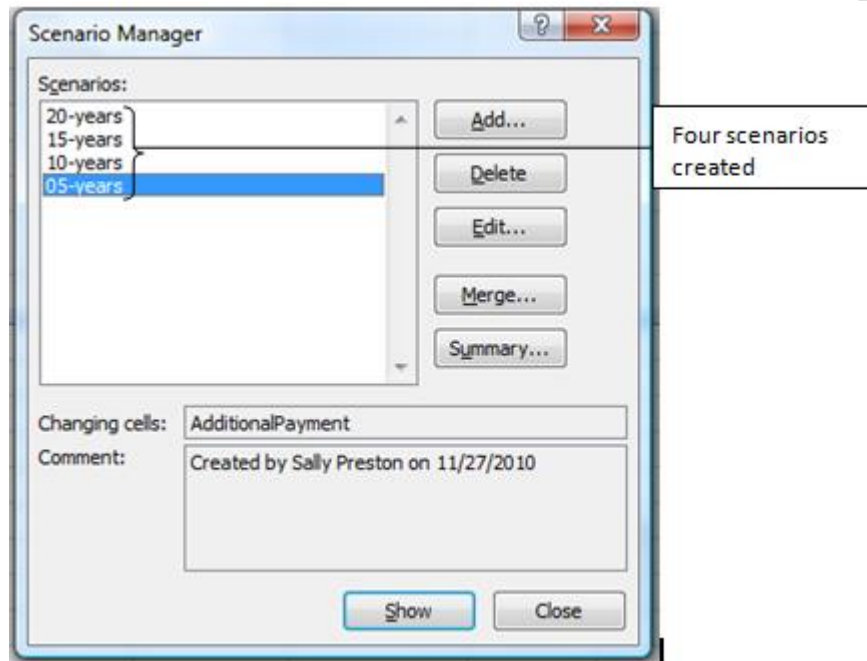


Figure 11.21. Four alternatives are saved using in Scenario Manager.

13. With the Scenario Manager dialog displayed, click the Summary button. In the Scenario Summary dialog box, confirm that Scenario summary is selected, and then in the *Results cells* box type **Total_Interest**. Recall that early you entered a sum function in cell B252 and named it Total_Interest.
14. Click OK. A new worksheet is created with a summary of the four options showing the scenario names as column headings, the changing cell—Additional_Payment, and the results cell—Total_Interest. Compare your screen with Figure 11.22. Notice how the total interest declines as the additional payment increases. Even paying off the loan five years sooner, in 2025, will save over \$18,000 in interest (\$66K-\$48K).

Save your work.

	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								

Scenario Summary					
	Current Values:	20-years	15-years	10-years	05-years
Changing Cells:					
Additional_Payment	(\$1,475.36)	\$0.00	(\$157.93)	(\$482.70)	(\$1,475.36)
Result Cells:					
Total_Interest	15,049.77	66,112.40	48,011.69	30,980.75	15,049.77

Notes: Current Values column represents values of changing cells at time Scenario Summary Report was created. Changing cells for each scenario are highlighted in gray.

Figure 11.22. The four scenarios are compared side-by-side in the Scenario Summary Report.

Use Goal Seek to Find the Internal Rate of Return (IRR)

The IRR worksheet displays a cash flow and investment analysis for a home improvement investment that would cost \$1,000 and save \$200 a year for nine years. The net present value (NPV) depends on how much the future savings are discounted. Recall that the discount rate is determined by considering how much it would cost the company or person to borrow the money or divert it from some other purpose. If money is expensive to obtain, the discount rate is higher. Using Goal Seek, you can determine how high the discount rate would have to be for the value of the future savings to equal the initial expense. This discount rate is called the **Internal Rate of Return (IRR)**. If the IRR for an investment is higher than the discount rate, it means that the investment will be worth the investment. If a choice is to be made between investments, the one with the higher IRR is preferred.

1. In Ch11OptimizeStudentName workbook, click the IRR sheet tab. Click cell B21. Change the discount rate to **0**. Observe that the net present value becomes \$800 which is the simple difference between the initial expense of \$1,000 and the \$1,800 saved in nine years.
2. In cell B21, change the discount rate to **3%**. Observe that the NPV value in B22 is reduced to \$540.99 because the future savings are worth less.
3. Click the What-If Analysis button and then click Goal Seek. In the dialog box, in the Set cell box, type

B22. In the *To value* box type **0**, and in the *By changing cell* box type **B21**. Click OK twice. The result is 14% which is the IRR for this investment. This is how much the future savings would have to be discounted to equal the initial expense. If individuals are considering this investment, they should compare it with other uses for the money. For example, if they have credit card debt on which they are paying 20% interest, they should use the \$1,000 to pay off the credit card debt because the discount rate they should use is 20% which is more than 14% for this investment.

4. In cell B21, type **9%** to return the value to its previous number.
5. In cell A23 type **Internal Rate of Return**. In B23, type **=IRR(B15:K15)**. This is the same range used by the NPV function and calculates the IRR for a series of future payments. The result is 14%, the same as the discount rate that was determined when Goal Seek found the discount rate for which the NPV was zero. This project would show a positive NPV if the cost of money is less than 14%. Some companies use an IRR value as a threshold for considering projects. Save your work.

Use Solver for Optimizing Several Variables

One of the advanced functions that is very useful but little known is called Solver. It is available as an **Add-in**, which means that it must be activated to show up on the Ribbon. Solver is a more powerful version of Goal Seek. Like Goal Seek, it focuses on one cell as the result but it can have several changing cells and each cell or range of cells can have criteria. Instead of a single output value, it can find minimums or maximums.

To find the best combination of inputs, the program tries different values in the changing cells. A spreadsheet can be designed several ways to use this function but the most important part is deciding how to use the changing cells.

Assigning Contracts

Solver can be used to assign contracts. With this approach, a model is used that shows the bids by several companies for several projects. In this model, Solver is used to assign a 1 or 0 into a range of changing cells

to indicate which contracts are awarded to which bidders. The SUMPRODUCT function is used to multiply the bids by the range of zeros and ones. The cell that is the output is the one that shows the total cost of all of the awarded bids. Typically the goal is to minimize the total costs. Business rules can be applied as constraints to the model.

On the Contracts worksheet you use Solver to determine the combination of bids that will result in the lowest total cost.

1. In Ch11OptimizeStudentName workbook, click the Contracts sheet tab. Observe the business rules in cells B14:B15. Observe that Wind Towers did not bid on the transformers contract as indicated by the empty cell E6.
2. Click cell F9. Use the SUM function to sum the range above; F5:F8 and then fill across to cell H9. After Solver is run, there will be a one or a zero in cells F5:H8 to indicate which company received the contract. The sum function in cells F9:H9 is used to be sure that each contract is only awarded to one company.
3. In cell I5, sum the contracts awarded in F5:H5, and then fill downward to cell I8.
4. With cells I5:I8 selected, click the Error Alert button that displays next to the cells, and then click Ignore Error to remove the green triangles that displayed in some of the cells. Recall that Excel will warn you when it thinks there may be an error in a formula. In this case, the adjacent cells with numbers in column E should not be included in the SUM function. The formula in column I is used to control how many contracts are awarded to each company.
5. In cell F10, type **=SUMPRODUCT(C5:C8,F5:F8)** and compare your screen with Figure 11.23. This function multiplies the bids in C5:C8 by the zeros or 1s in F5:F8 and then sums the result to determine the cost of the bid awarded for the turbine project. Press Enter to complete the function.

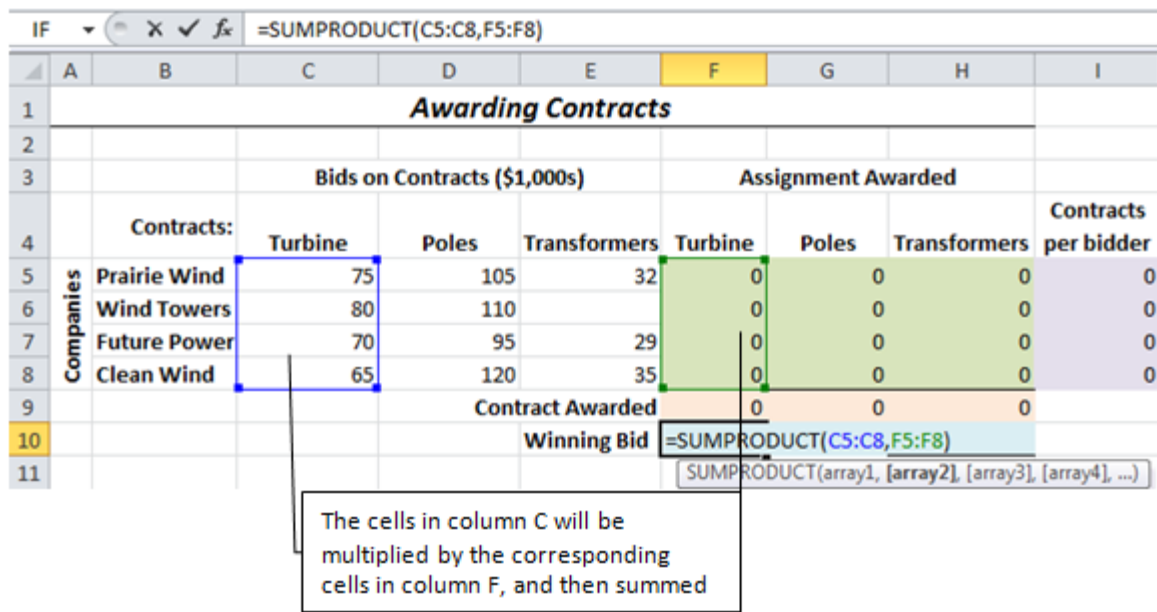


Figure 11.23. The SUMPRODUCT function is used to multiply two columns of numbers and sum the results.

6. Click cell F10. On the Formulas tab, in the Formula Auditing group, click the Trace Precedents button. Cells C5:C8 and F5:F8 are outlined and two arrows display to show that these ranges are used in the results displayed in cell F10. This feature is useful to show which cells or ranges are used by a function.
7. In the Formula Auditing group, click the Remove Arrows button.
8. In cell G10, type **=SUMPRODUCT(D5:D8,G5:G8)** and click the Enter button . This multiplies the bids in D5:D8 by the zeros or 1s that will display in G5:G8. In the Formula Auditing group click the Trace Precedents arrows to show the cells used in this formula. Compare your screen to Figure 11.24.

15. Click cell H11 to make it the active cell. On the Data tab, in the Analysis group, click the Solver button.
- In the Solver dialog box, be sure the Set Objective cell displays \$H\$11, and then click the Min option button. The goal is to minimize the total cost of the contracts that are awarded.
16. In the By Changing Variable Cells box type **F5:H8**. This is the range of cells that will identify the contracts that will be assigned—the green area on the worksheet.
17. Next to the Subject to the Constraints box click the Add button. The first constraint for the first business rule—shown in pink—is to limit the contracts so that only one bid is accepted for each of the contracts. In the Add Constraint dialog box, in the Cell Reference box type **F9:H9**. In the middle box, click the arrow and then click **=**. In the Constraint box type **1**. Compare your dialog box to Figure 11.25.

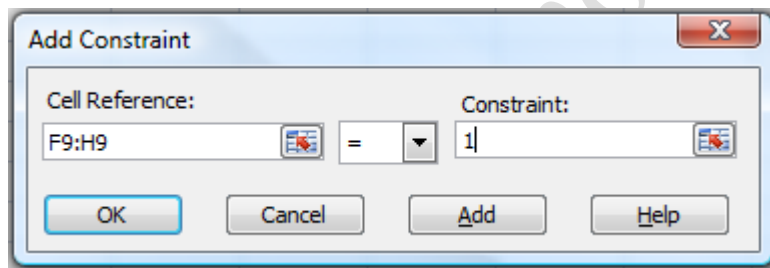


Figure 11.25. The total for each contract must be 1.

18. Click Add. The current constraint is saved and the dialog box is cleared so that you can add another constraint. In the Cell Reference box type **I5:I8**. In the middle box select **<=** and in the Constraint box type **1**. This constraint limits the contracts so that no more than one contract is awarded to a bidder. This is the second business rule—shown in light purple. Click Add.
19. To add the third constraint, in the Cell Reference dialog box type **F5:H8**. In the middle box, click the arrow and then click bin. This constraint will force the values in the changing cells to be either of the binary numerals, 0 or 1.
20. Click OK and compare your Solver Parameters dialog box with Figure 11.26.

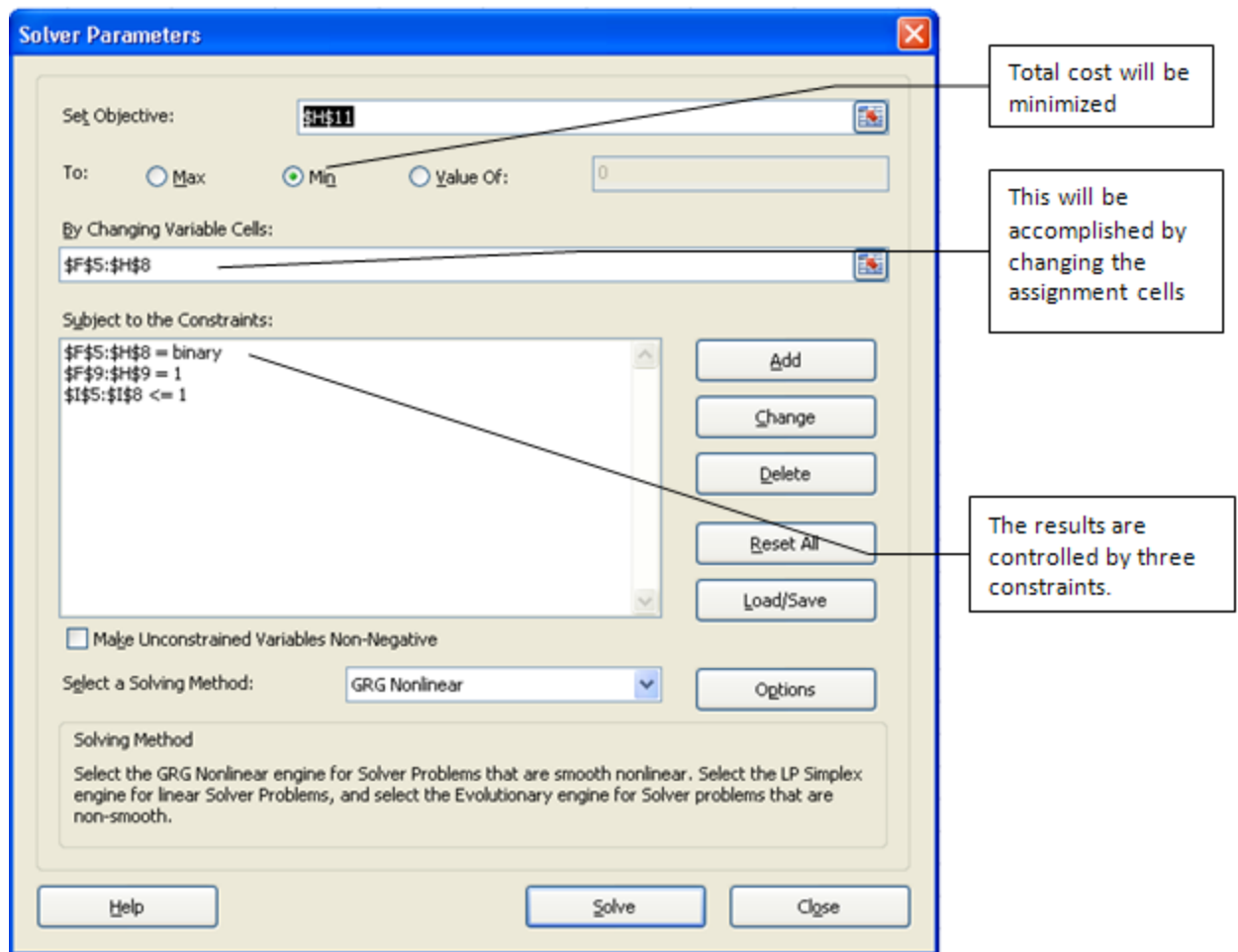


Figure 11.26. Target cell, changing cells, and constraints identified in Solver.

21. Click the Solve button to find a solution. Look at the results and notice that the program awarded the contract for the Transformers to Wind Towers who did not submit a bid for that contract. The amount in cell H10 for the Transformer contract is zero. While this is certainly the minimum, the computer does not have the world knowledge to know that a blank means no bid rather than the willingness to work for nothing.
22. In the Solver Results dialog box click Cancel. Another constraint is needed.
23. In the Analysis group click Solver to redisplay the Solver Parameters dialog box. Click the Add button. In the Add Constraint dialog box, enter a constraint so that that F10:H10 \geq 1 and then click OK.
24. Click Solve again. The Target cell—Total Costs— should be 192 (thousand). Click OK to accept the

solution.

25. Click cell B17 and type 4) **Winning bid ≥ 1** . Merge cells B17:D17 and align the text on the left. Add a Fill Color of *Aqua Accent 5, Lighter 80%*. Apply the same Fill Color to F10:H10 where this rule applies.
26. Save your work and compare your results with Figure 11.27.

	A	B	C	D	E	F	G	H	I
1	Awarding Contracts								
2									
3			Bids on Contracts (\$1,000s)			Assignment Awarded			
4		Contracts:	Turbine	Poles	Transformers	Turbine	Poles	Transformers	Contracts per bidder
5	Companies	Prairie Wind	75	105	32	0	0	1	1
6		Wind Towers	80	110		0	0	0	0
7		Future Power	70	95	29	0	1	0	1
8		Clean Wind	65	120	35	1	0	0	1
9					Contract Awarded	1	1	1	
10					Winning Bid	65	95	32	
11						Total Costs		192	
12									
13		Business Rules:							
14		1) Only one bidder per contract							
15		2) One contract per bidder							
16		3) Changing cells are binary							
17		4) Winning Bid ≥ 1							
18		Goal: Minimize the total cost of the contracts.							

Constraints control the outcome

The changing cells display a 1 or 0 to indicate which company was awarded which contract.

Total Cost was minimized

Figure 11.27. Solver assigns the contracts to minimize the costs and insure one bidder per contract, and only one contract goes to each company.

Assigning Staff to Schedules

To use Solver to assign staff to a work schedule, a model is used that is similar to the model used for assigning contracts except in this model, the grid of 1s and 0s is part of the model where it is used to indicate shift assignments. Solver is used to assign the number of staff to each schedule. Empty cells can be used instead of zeros because the SUMPRODUCT function will treat an empty cell like a zero.

On the Personnel worksheet you will use Solver to determine the combination of staff and schedules that

meets minimum staffing needs each day and minimizes the extra staff days. This example has two business rules. They are:

1) The total staff assigned cannot exceed the staff available

2) The number of staff assigned to each day must not be less than the minimum required for that day.

1. In Ch11OptimizeStudentName workbook, click the Personnel sheet tab. Observe that seven work shifts are listed in column A and the Days Off for each work shift is indicated in column B. Observe that 15 staff members are available, as indicated in B2.
2. In cells E5:I5 enter **1** to indicate the days that are workdays for Shift A—Monday through Friday. Cells C5—Sunday—and I5—Saturday—are left blank because they are the days off for Shift A.
3. Continue to enter a **1** for each work-day in rows 6 through 11. Refer to the days off in column B for each shift. In the cells to the right in columns D through J, leave those cells empty for the days off and fill the others by typing **1** to indicate a work-day. Compare your results with Figure 11.28.

	A	B	C	D	E	F	G	H	I	J
1	Personnel Scheduling									
2	Staff Available	15								
3				1=workday, blank=day off						
4	Shift	Days Off	Staff Assigned	Sun	Mon	Tue	Wed	Thu	Fri	Sat
5	A	Sat Sun			1	1	1	1	1	
6	B	Sun Mon				1	1	1	1	1
7	C	Mon Tues		1			1	1	1	1
8	D	Tues Wed		1	1			1	1	1
9	E	Wed Thur		1	1	1			1	1
10	F	Thur Fri		1	1	1	1			1
11	G	Fri Sat		1	1	1	1	1		
12	Staff Required									

Figure 11.28. The work schedule uses a 1 to represent a day worked and blank for days off.

4. Selected cells C5:C11. Apply the Fill Color: *Olive Green, Accent 3, Lighter 60%*. These are the changing cells—the ones that will be filled by Solver to allocate the staff to each work shift.
5. In cell C12, type **=SUM(C5:C11)** to find the total number of employees assigned to the work schedule.

6. In D13:J13 type 12, 6, 7, 12, 8, 9, and 10. This is the minimum staff required for each day listed in row 4—Sunday through Saturday.
7. Select cells C5:C11. On the Formula Bar, in the Name box, type Staff_per_Shift to give this range a name.
8. In D14, type =SUMPRODUCT(Staff_per_Shift,D5:D11) and then press Enter. The SUMPRODUCT function determines the number of people assigned to work on Sundays. The number assigned by Solver in column C will be multiplied by the 1 or blank in column D and then summed. The range named Staff_per_Shift is creates an absolute reference so that it can be filled across the row to calculate the sum the employees assigned to work each day of the week.
9. Click cell D14 and use the fill handle to fill this formula across to cell J14. Currently zero displays as the result because cells C5:C11 are empty.
10. In cell D15, type =D14-D13. Fill the formula across to cell J15. This will show when there is more or less staff than needed each day. Currently these values are all negative because no staff has been assigned to the shifts.
11. In cell J16, type =SUM(D15:J15). This is the target cell—the goal is to minimize extra staff days but this total should be zero or positive.
12. Click cell C12. On the Home tab, in the Styles group, click the Conditional Formatting button. Point to Highlight Cells Rules and then click Greater Than. Click in the first box and then click cell B2. In the second box, click the arrow and then click to select Red Text. Click OK. If the staff assigned is more than the staff available, then cell B12 will display with red text.
13. Use the Conditional Formatting Style to apply a red text format to cells D15:J15 if they are <0.
14. On the Data tab, click the Solver button. Set cell J16 as the Target cell and select Min. In the *By Changing Variable Cells* box type Staff_per_Shift
15. Click the Add button and enter a constraint to ensure that the Staff required does not exceed the staff

available: $C12 \leq B2$

16. Add another constraint so that $D14:J14 \geq D13:J13$. Two ranges of equal size can be compared pair-wise, where the first number in the first range is \geq to the first number in the second range, the second number in the first range is \geq to the second number in the second range and so forth.
17. Click OK in the Add Constraint dialog box. In the Solver Parameters dialog box click the Solve button to run Solver. Observe that world knowledge is needed to constrain the answers in column C to whole numbers of staff.
18. Cancel the Solver results. Click the Solver button and Add a constraint so that Staff_per_Shift are integers. Use the middle box to select *int* from the list. Compare your dialog box with Figure 11.29.



Figure 11.29. The number of staff assigned is limited to whole numbers.

19. Click OK. Confirm that the Make Unconstrained Variables Non-negative box is checked to prevent assigning negative staff.

Tip: If you are using a version of Excel prior to 2010, the check box in step 19 does not exist. Instead it is necessary to add a constraint for the changing cells to be >0 to ensure that the results do not include negative numbers.

20. Run Solver two or three times. Observe there is more than one solution with 6 extra staff days. Each of them could be saved as a scenario and discussed at a staff meeting to decide between options.
21. Save your work with one of the solutions displayed.

Key Takeaways

- Goal Seek is used to see the effect of changing one input on a particular output. In a loan repayment schedule

it can be used to determine the amount of additional payment required to pay-off the loan early and the alternatives can be saved as scenarios for side-by-side comparisons. [11.2.1]

- The internal rate of return is the discount value that makes the net present value equal zero. Use Goal Seek to find that discount rate by choosing the NPV cell as the output, setting it to zero, and then changing the discount rate. The resulting discount rate is the IRR. Alternatively, use the IRR function and specify the same range of cells used by the NPV function. [11.2.2]
- The Solver function has several arguments. Like Goal Seek, it has one output target cell that can be matched or minimized or maximized. A group of changing cells is selected to determine the output. Constraints are added that meet business rules. World knowledge is applied with additional constraints. In the contract assignment model, the binary constraint requires the numbers chosen to be zero or one. In the staffing model, the constraint of Integer is used to require choice of whole people and positive numbers. [11.2.3]
- Precedent and dependent arrows can be used to show which cells are used by a formula or which formulas use the current cell. [11.2.4]

Key Terms

Add-in

Additional programs that must be activated to appear on the ribbon.

Assignment model

Computer model where the the computer determines where resources should go.

Changing cell

Location of an independent variable that can be altered to produce different results in dependent formulas.

Computer modeling

An interactive set of formulas that behave in a manner that is similar to a real system.

Constraint

Limiting condition similar to a criteria.

Feasible

Possible

Feedback

The output is used to adjust the input.

Integer

Whole numbers including negatives.

Internal Rate of Return (IRR)

Discount rate at which the net present value is zero.

Scenario

Set of input values that are saved with a name to be compared with other sets of input values.

Sensitivity analysis

Determining how much the output changes in response to changes in an input.

Solver

Program that calculates a desired outcome including maximum and minimum values based on several inputs and constraints.

Works Cited

Kakley, Matt. *The Sun Chronicle*. November 18, 2008. <http://www.thesunchronicle.com/articles/2008/11/18/news/3949473.txt> (accessed October 29, 2010).

Tehran Times International Daily. *Society*. November 20, 2008. <http://www.tehrantimes.com/PDF/10466/10466-12.pdf> (accessed October 28, 2010).

Last edited on February 21, 2011