Homework 3
TIM 206, Fall 2011
Due: Tuesday, Nov 15, in class.

Reading: Chapters 10, 11, 13 Hillier and Lieberman
Problems identified by number are taken from Hillier and Lieberman 8th edition. (Note this may not be the edition in the library).

1. Problem 10.3-3
2. Problem 10.4-2
3. Problem 11.3-3
4. Problem 11.6-1
5. Problem 13.1-1
6. Problem 13.2-4
7. Problem 13.3-3(a)
8. Problem 13.4-8(a)
   (use any random number generator but show your working)
10.2-4. Consider the following statements about solving dynamic programming problems. Label each statement as true or false, and then justify your answer by referring to specific statements (with page citations) in the chapter.

(a) The solution procedure uses a recursive relationship that enables solving for the optimal policy for stage \((n + 1)\) given the optimal policy for stage \(n\).

(b) After completing the solution procedure, if a nonoptimal decision is made by mistake at some stage, the solution procedure will need to be reapplied to determine the new optimal decisions (given this nonoptimal decision) at the subsequent stages.

(c) Once an optimal policy has been found for the overall problem, the information needed to specify the optimal decision at a particular stage is the state at that stage and the decisions made at preceding stages.

10.3-1. The owner of a chain of three grocery stores has purchased five crates of fresh strawberries. The estimated probability distribution of potential sales of the strawberries before spoilage differs among the three stores. Therefore, the owner wants to know how to allocate five crates to the three stores to maximize expected profit.

For administrative reasons, the owner does not wish to split crates between stores. However, he is willing to distribute no crates to any of his stores.

The following table gives the estimated expected profit at each store when it is allocated various numbers of crates:

<table>
<thead>
<tr>
<th>Crates</th>
<th>Store 1</th>
<th>Store 2</th>
<th>Store 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>11</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>14</td>
<td>15</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>17</td>
<td>19</td>
<td>18</td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td>22</td>
<td>20</td>
</tr>
</tbody>
</table>

Use dynamic programming to determine how many of the five crates should be assigned to each of the three stores to maximize the total expected profit.

10.3-2. A college student has 7 days remaining before final examinations begin in her four courses, and she wants to allocate this study time as effectively as possible. She needs at least 1 day on each course, and she likes to concentrate on just one course each day, so she wants to allocate 1, 2, 3, or 4 days to each course. Having recently taken an OR course, she decides to use dynamic programming to make these allocations to maximize the total grade points to be obtained from the four courses. She estimates that the alternative allocations for each course would yield the number of grade points shown in the following table:

Solve this problem by dynamic programming.

10.3-3. A political campaign is entering its final stage, and polls indicate a very close election. One of the candidates has enough funds left to purchase TV time for a total of five prime-time commercials on TV stations located in four different areas. Based on polling information, an estimate has been made of the number of additional votes that can be won in the different broadcasting areas depending upon the number of commercials run. These estimates are given in the following table in thousands of votes:

<table>
<thead>
<tr>
<th>Commercials</th>
<th>Area 1</th>
<th>Area 2</th>
<th>Area 3</th>
<th>Area 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>6</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>11</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>12</td>
<td>9</td>
<td>9</td>
</tr>
</tbody>
</table>

Use dynamic programming to determine how the five commercials should be distributed among the four areas in order to maximize the estimated number of votes won.

10.3-4. A county chairman of a certain political party is making plans for an upcoming presidential election. She has received the services of six volunteer workers for precinct work, and she wants to assign them to four precincts in such a way as to maximize their effectiveness. She feels that it would be inefficient to assign a worker to more than one precinct, but she is willing to assign no workers to any one of the precincts if they can accomplish more in other precincts.

The following table gives the estimated increase in the number of votes for the party's candidate in each precinct if it were located various numbers of workers:

<table>
<thead>
<tr>
<th>Workers</th>
<th>Precinct 1</th>
<th>Precinct 2</th>
<th>Precinct 3</th>
<th>Precinct 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>7</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>11</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>16</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>4</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>5</td>
<td>22</td>
<td>20</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>6</td>
<td>24</td>
<td>21</td>
<td>22</td>
<td>22</td>
</tr>
</tbody>
</table>
Because of budget limitations, a maximum of $1,000 can be expended. Use dynamic programming to determine how many parallel units should be installed in each of the four components to maximize the probability that the system will function.

10.3-8. Consider the following integer nonlinear programming problem.

Maximize \[ Z = 3x_1^2 - x_1^3 + 5x_2^2 - x_2, \]
subject to
\[ x_1 + 2x_2 \leq 4 \]
and
\[ x_1 \geq 0, \quad x_2 \geq 0 \]
\[ x_1, x_2 \text{ are integers.} \]

Use dynamic programming to solve this problem.

10.3-9. Consider the following integer nonlinear programming problem.

Maximize \[ Z = 18x_1 - x_1^2 + 20x_2 + 10x_3, \]
subject to
\[ 2x_1 + 4x_2 + 3x_3 \leq 11 \]
and
\[ x_1, x_2, x_3 \text{ are nonnegative integers.} \]

Use dynamic programming to solve this problem.

10.3-10. Consider the following nonlinear programming problem.

Maximize \[ Z = 36x_1 + 9x_1^2 - 6x_1^3 + 36x_2 - 3x_3^2, \]
subject to
\[ x_1 + x_2 \leq 3 \]
and
\[ x_1 \geq 0, \quad x_2 \geq 0. \]

Use dynamic programming to solve this problem.

10.3-11. Re-solve the Local Job Shop employment scheduling problem (Example 4) when the total cost of changing the level of employment from one season to the next is changed to $100 times the square of the difference in employment levels.

10.3-12. Consider the following nonlinear programming problem.

Maximize \[ Z = 2x_1^2 + 2x_2 + 4x_3 - x_3 \]
subject to
\[ 2x_1 + x_2 + x_3 \leq 4 \]
and
\[ x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0. \]

Use dynamic programming to solve this problem.

10.3-13. Consider the following nonlinear programming problem.

Minimize \[ Z = x_1^4 + 2x_1^2 \]
subject to
\[ x_1^3 + x_2^3 \geq 2. \]
(There are no nonnegativity constraints.) Use dynamic programming to solve this problem.

10.3-14. Consider the following nonlinear programming problem.

Maximize \[ Z = x_1^4 + 4x_2^3 + 16x_3, \]
subject to
\[ x_1x_2x_3 = 4 \]
and
\[ x_1 \geq 1, \quad x_2 \geq 1, \quad x_3 \geq 1. \]
(a) Solve by dynamic programming when, in addition to the given constraints, all three variables also are required to be integer.
(b) Use dynamic programming to solve the problem as given (continuous variables).

10.3-15. Consider the following nonlinear programming problem.

Maximize \[ Z = x_1(1 - x_2)x_3, \]
subject to
\[ x_1 - x_2 + x_3 \leq 1 \]
and
\[ x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0. \]

Use dynamic programming to solve this problem.

10.3-16. Consider the following linear programming problem.

Maximize \[ Z = 15x_1 + 10x_2, \]
subject to
\[ x_1 + 2x_2 \leq 6 \]
\[ 3x_1 + x_2 \leq 8 \]
and
\[ x_1 \geq 0, \quad x_2 \geq 0. \]

Use dynamic programming to solve this problem.

10.3-17. Consider the following "fixed-charge" problem.

Maximize \[ Z = 3x_1 + 7x_2 + 6f(x_3), \]
subject to
\[ x_1 + 3x_2 + 2x_3 \leq 6 \]
\[ x_1 \geq 0, \quad x_2 \leq 5 \]
and
\[ x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0. \]
11.1-7. Reconsider Prob. 8.2-21 involving a contractor (Susan Meyer) who needs to arrange for hauling gravel from two pits to three building sites.

Susan now needs to hire the trucks (and their drivers) to do the hauling. Each truck can only be used to haul gravel from a single pit to a single site. In addition to the hauling and gravel costs specified in Prob. 8.2-21, there now is a fixed cost of $50 associated with hiring each truck. A truck can haul 5 tons, but it is not required to go full. For each combination of pit and site, there are now two decisions to be made: the number of trucks to be used and the amount of gravel to be hauled.
(a) Formulate an MIP model for this problem.
(b) Use the computer to solve this model.

11.2-1. Select one of the actual applications of BIP by a company or governmental agency mentioned in Sec. 11.2. Read the article describing the application in the referenced issue of Interfaces. Write a two-page summary of the application and its benefits.

11.2-2. Select three of the actual applications of BIP by a company or governmental agency mentioned in Sec. 11.2. Read the articles describing the applications in the referenced issues of Interfaces. For each one, write a one-page summary of the application and its benefits.

11.3-1.* The Research and Development Division of the Progressive Company has been developing four possible new product lines. Management must now make a decision as to which of these four products actually will be produced and at what levels. Therefore, an operations research study has been requested to find the most profitable product mix.

A substantial cost is associated with beginning the production of any product, as given in the first row of the following table. Management’s objective is to find the product mix that maximizes the total profit (total net revenue minus start-up costs).

<table>
<thead>
<tr>
<th>Product</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start-up cost</td>
<td>$50,000</td>
<td>$40,000</td>
<td>$70,000</td>
<td>$60,000</td>
</tr>
<tr>
<td>Marginal revenue</td>
<td>$70</td>
<td>$70</td>
<td>$60</td>
<td>$90</td>
</tr>
</tbody>
</table>

Let the continuous decision variables \( x_1, x_2, x_3, \) and \( x_4 \) be the production levels of products 1, 2, 3, and 4, respectively. Management has imposed the following policy constraints on these variables:

1. No more than two of the products can be produced.
2. Either product 3 or 4 can be produced only if either product 1 or 2 is produced.
3. Either \( 5x_1 + 3x_2 + 6x_3 + 4x_4 \leq 6,000 \)
   or \( 4x_1 + 6x_2 + 3x_3 + 5x_4 \leq 6,000 \).

(a) Introduce auxiliary binary variables to formulate a mixed BIP model for this problem.
(b) Use the computer to solve this model.

11.3-2. Suppose that a mathematical model fits linear programming except for the restriction that \( |x_1 - x_2| = 0, 3, \) or 6. Show how to reformulate this restriction to fit an MIP model.

11.3-3. Suppose that a mathematical model fits linear programming except for the restrictions that:

1. At least one of the following two inequalities holds:
   \[ x_1 + x_2 + x_3 + x_4 \leq 4 \]
   \[ 3x_1 - x_2 - x_3 + x_4 \leq 3. \]

2. At least two of the following four inequalities hold:
   \[ 5x_1 + 3x_2 + 3x_3 - x_4 \leq 10 \]
   \[ 2x_1 + 5x_2 - x_3 + 3x_4 \leq 10 \]
   \[ -x_1 + 3x_2 + 5x_3 + 3x_4 \leq 10 \]
   \[ 3x_1 - x_2 + 3x_3 + 5x_4 \leq 10. \]

Show how to reformulate these restrictions to fit an MIP model.

11.3-4. The Toys-R-4-U Company has developed two new toys for possible inclusion in its product line for the upcoming Christmas season. Setting up the production facilities to begin production would cost $50,000 for toy 1 and $80,000 for toy 2. Once these costs are covered, the toys would generate a unit profit of $10 for toy 1 and $15 for toy 2.

The company has two factories that are capable of producing these toys. However, to avoid doubling the start-up costs, just one factory would be used, where the choice would be based on maximizing profit. For administrative reasons, the same factory would be used for both new toys if both are produced.

Toy 1 can be produced at the rate of 50 per hour in factory 1 and 40 per hour in factory 2. Toy 2 can be produced at the rate of 40 per hour in factory 1 and 25 per hour in factory 2. Factories 1 and 2, respectively, have 500 hours and 700 hours of production time available before Christmas that could be used to produce these toys.

It is not known whether these two toys would be continued after Christmas. Therefore, the problem is to determine how many units (if any) of each new toy should be produced before Christmas to maximize the total profit.

(a) Formulate an MIP model for this problem.
(b) Use the computer to solve this model.

11.3-5.* Northeastern Airlines is considering the purchase of new long-, medium-, and short-range jet passenger airplanes. The purchase price would be $67 million for each long-range plane, $51 million for each medium-range plane, and $35 million for each short-range plane. The board of directors has authorized a maximum commitment of $1.5 billion for these purchases. Regardless of which airplanes are purchased, air travel of all distances is expected to be sufficiently large that these planes would be utilized at essentially maximum capacity. It is estimated that the net annual profit (after capital recovery costs are subtracted) would be $42 million per long-range plane, $3 million per medium-range plane, and $2.3 million per short-range plane.

It is predicted that enough trained pilots will be available to the company to crew 30 new airplanes. If only short-range planes...
11.5.3. Follow the instructions of Prob. 11.5-1 for the following BIP problem.

\[ \text{Maximize} \quad Z = 2x_1 + 5x_2, \]

subject to
\[ 10x_1 + 30x_2 \leq 30 \]
\[ 95x_1 - 30x_2 \leq 75 \]

and
\[ x_1, x_2 \text{ are binary.} \]

11.5.4. Follow the instructions of Prob. 11.5-1 for the following BIP problem.

\[ \text{Maximize} \quad Z = -5x_1 + 25x_2, \]

subject to
\[ -3x_1 + 30x_2 \leq 27 \]
\[ 3x_1 + x_2 \leq 4 \]

and
\[ x_1, x_2 \text{ are binary.} \]

11.5.5. Label each of the following statements as True or False, and then justify your answer by referring to specific statements (with page citations) in the chapter.

(a) Linear programming problems are generally much easier to solve than IP problems.
(b) For IP problems, the number of integer variables is generally more important in determining the computational difficulty than is the number of functional constraints.
(c) To solve an IP problem with an approximate procedure, one may apply the simplex method to the LP relaxation problem and then round each noninteger value to the nearest integer. The result will be a feasible but not necessarily optimal solution for the IP problem.

11.6-1. Use the BIP branch-and-bound algorithm presented in Sec. 11.6 to solve the following problem interactively.

\[ \text{Maximize} \quad Z = 2x_1 - x_2 + 5x_3 - 3x_4 + 4x_5, \]

subject to
\[ 3x_1 - 2x_2 + 7x_3 - 5x_4 + 4x_5 \leq 6 \]
\[ x_1 - x_2 + 2x_3 - 4x_4 + 2x_5 \leq 0 \]

and
\[ x_j \text{ is binary, for } j = 1, 2, \ldots, 5. \]

11.6-2. Use the BIP branch-and-bound algorithm presented in Sec. 11.6 to solve the following problem interactively.

\[ \text{Minimize} \quad Z = 5x_1 + 6x_2 + 7x_3 + 8x_4 + 9x_5, \]

subject to
\[ 3x_1 - x_2 + x_3 + x_4 - 2x_5 \geq 2 \]
\[ x_1 + 3x_2 - x_3 - 2x_4 + x_5 \geq 0 \]
\[ -x_1 - x_2 + 3x_3 + x_4 + x_5 \geq 1 \]

and
\[ x_j \text{ is binary, for } j = 1, 2, \ldots, 5. \]

11.6-3. Use the BIP branch-and-bound algorithm presented in Sec. 11.6 to solve the following problem interactively.

\[ \text{Maximize} \quad Z = 5x_1 + 5x_2 + 8x_3 - 2x_4 - 4x_5, \]

subject to
\[ -3x_1 + 6x_2 - 7x_3 + 9x_4 + 9x_5 \geq 10 \]
\[ x_1 + 2x_2 - x_4 - 3x_5 \leq 0 \]

and
\[ x_j \text{ is binary, for } j = 1, 2, \ldots, 5. \]

11.6-4. Reconsider Prob. 11.3-6(a). Use the BIP branch-and-bound algorithm presented in Sec. 11.6 to solve this BIP model interactively.

11.6-5. Reconsider Prob. 11.4-10(a). Use the BIP algorithm presented in Sec. 11.6 to solve this problem interactively.

11.6-6. Consider the following statements about any pure IP problem (in maximization form) and its LP relaxation. Label each of the statements as True or False, and then justify your answer.

(a) The feasible region for the LP relaxation is a subset of the feasible region for the IP problem.
(b) If an optimal solution for the LP relaxation is an integer solution, then the optimal value of the objective function is the same for both problems.
(c) If a noninteger solution is feasible for the LP relaxation, then the nearest integer solution (rounding each variable to the nearest integer) is a feasible solution for the IP problem.

11.6-7. Consider the assignment problem with the following cost table:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>39</td>
<td>65</td>
<td>69</td>
<td>66</td>
<td>57</td>
</tr>
<tr>
<td>2</td>
<td>64</td>
<td>84</td>
<td>24</td>
<td>92</td>
<td>22</td>
</tr>
<tr>
<td>Assignee</td>
<td>3</td>
<td>49</td>
<td>50</td>
<td>61</td>
<td>31</td>
</tr>
<tr>
<td>4</td>
<td>48</td>
<td>45</td>
<td>55</td>
<td>23</td>
<td>50</td>
</tr>
<tr>
<td>5</td>
<td>59</td>
<td>34</td>
<td>30</td>
<td>34</td>
<td>18</td>
</tr>
</tbody>
</table>

(a) Design a branch-and-bound algorithm for solving such assignment problems by specifying how the branching, bounding, and fathoming steps would be performed. (Hint: For the assignees not yet assigned for the current subproblem, form the relaxation by deleting the constraints that each of these assignees must perform exactly one task.)
(b) Use this algorithm to solve this problem.

11.6-8. Five jobs need to be done on a certain machine. However, the setup time for each job depends upon which job immediately preceded it, as shown by the following table:
PROBLEMS

The symbol A to the left of some of the problems (or their parts) has the following meaning.

A: You should use the corresponding automatic procedure in IOR Tutorial. The printout will record the results obtained at each iteration.

An asterisk on the problem number indicates that at least a partial answer is given in the back of the book.

Instructions for Obtaining Random Numbers

For each problem or its part where random numbers are needed, obtain them from the consecutive random digits in Table 20.3 in Sec. 20.3 as follows. Start from the front of the top row of the table and form five-digit random numbers by placing a decimal point in front of each group of five random digits (0.09656, 0.96567, etc.) in the order that you need random numbers. Always restart from the front of the top row for each new problem or its part.

13.1.1. Consider the traveling salesman problem shown below, where city 1 is the home city.

(a) List all the possible tours, except exclude those that are simply the reverse of previously listed tours. Calculate the distance of each of these tours and thereby identify the optimal solution.
(b) Starting with 1-2-3-4-5-6-1 as the initial trial solution, apply the sub-tour reversal algorithm to this problem.
(c) Apply the sub-tour reversal algorithm to this problem when starting with 1-2-3-4-5-6-1 as the initial trial solution.

13.1.2. Reconsider the example of a traveling salesman problem shown in Fig. 13.4.

(a) When the sub-tour reversal algorithm was applied to this problem in Sec. 13.1, the first iteration resulted in a tie for which of two sub-tour reversals (reversing 3-4 or 4-5) provided the largest decrease in the distance of the tour, so the tie was broken arbitrarily in favor of the first reversal. Determine what would have happened if the second of these reversals (reversing 3-5) had been chosen instead.
(b) Apply the sub-tour reversal algorithm to this problem when starting with 1-2-4-5-6-7-3-1 as the initial trial solution.

13.1.3. Consider the traveling salesman problem shown below, where city 1 is the home city.

(a) List all the possible tours, except exclude those that are simply the reverse of previously listed tours. Calculate the distance of each of these tours and thereby identify the optimal solution.
(b) Starting with 1-2-3-4-5-6-1 as the initial trial solution, apply the sub-tour reversal algorithm to this problem.
(c) Apply the sub-tour reversal algorithm to this problem when starting with 1-2-5-4-3-6-1 as the initial trial solution.

13.2.1. Consider the minimum spanning tree problem depicted below, where the dashed lines represent the potential links that could be inserted into the network and the number next to each dashed line represents the cost associated with inserting that particular link.

This problem also has the following two constraints:

Constraint 1: No more than one of the three links—AB, BC, and AE—can be included.
Constraint 2: Link AB can be included only if link BD also is included.

Starting with the initial trial solution where the inserted links are AB, AC, AE, and CD, apply the basic tabu search algorithm presented in Sec. 13.2 to this problem.

13.2-2. Reconsider the example of a constrained minimum spanning tree problem presented in Sec. 13.2 (see Fig. 13.7(a) for the data before introducing the constraints). Starting with a different initial trial solution, namely, the one with links AB, AD, BE, and CD, apply the basic tabu search algorithm again to this problem.

13.2-3. Reconsider the example of an unconstrained minimum spanning tree problem given in Sec. 9.4. Suppose that the following constraints are added to the problem.

Constraint 1: Either link AD or link ET must be included.

Constraint 2: At most one of the three links—AO, BE, and DE—can be included.

Starting with the optimal solution for the unconstrained problem given at the end of Sec. 9.4 as the initial trial solution, apply the basic tabu search algorithm to this problem.

13.2-4. Reconsider the traveling salesman problem shown in Prob. 13.1-1. Starting with 1-2-4-3-5-1 as the initial trial solution, apply the basic tabu search algorithm by hand to this problem.

A 13.2-5. Consider the 8-city traveling salesman problem whose links have the associated distances shown in the following table (where a dash indicates the absence of a link).

<table>
<thead>
<tr>
<th>City</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14</td>
<td>15</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>13</td>
<td>14</td>
<td>20</td>
<td>—</td>
<td>—</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>21</td>
<td>17</td>
<td>9</td>
<td>9</td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>11</td>
<td>10</td>
<td>8</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>18</td>
<td>—</td>
<td></td>
<td>9</td>
<td></td>
<td>13</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

City 1 is the home city. Starting with each of the initial trial solutions listed below, apply the basic tabu search algorithm in your IOR Tutorial to this problem. In each case, count the number of times that the algorithm makes a nonimproving move. Also point out any tabu moves that are made anyway because they result in the best trial solution found so far.

(a) Use 1-2-3-4-5-6-7-8-1 as the initial trial solution.
(b) Use 1-2-5-6-7-4-8-3-1 as the initial trial solution.
(c) Use 1-3-2-5-6-4-7-8-1 as the initial trial solution.

13.3-1. While applying a simulated annealing algorithm to a certain problem, you have come to an iteration where the current value of \( T = 2 \) and the value of the objective function for the current trial solution is 30. This trial solution has four immediate neighbors and their objective function values are 29, 34, 31, and 24. For each of these four immediate neighbors in turn, you wish to determine the probability that the move selection rule would accept this immediate neighbor if it is randomly selected to become the current candidate to be the next trial solution.

(a) Determine this probability for each of the immediate neighbors when the objective is maximization of the objective function.
(b) Determine this probability for each of the immediate neighbors when the objective is minimization of the objective function.

A 13.3-2. Because of its use of random numbers, a simulated annealing algorithm will provide slightly different results each time it is run. Table 13.5 shows one application of the basic simulated annealing algorithm in IOR Tutorial to the example of a traveling salesman problem depicted in Fig. 13.4. Starting with the same initial trial solution (1-2-3-4-5-6-7-1), use your IOR Tutorial to apply the same algorithm to this same example five more times. How many times does it again find the optimal solution (1-3-5-7-6-4-2-1 or, equivalently, 1-2-4-6-7-5-3-1)?

13.3-3. Reconsider the traveling salesman problem shown in Prob. 13.1-1. Using 1-2-3-4-5-1 as the initial trial solution, you are to follow the instructions below for applying the basic simulated annealing algorithm presented in Sec. 13.3 to this problem.

(a) Perform the first iteration by hand. Follow the instructions given at the beginning of the Problems section to obtain the needed random numbers. Show your work, including the use of the random numbers.
Does this application again obtain the optimal solution \(x = 20\), just as was found during the first iteration in Table 13.7?

(b) Because of its use of random numbers, a genetic algorithm will provide slightly different results each time it is run. Use your IOR Tutorial to apply the basic genetic algorithm described in Sec. 13.4 to this same example five more times. How many times does it again find the optimal solution \(x = 20\)?

13.4-4. Reconsider the nonconvex programming problem shown in Prob. 13.3-7. Suppose now that the variable \(x\) is restricted to be an integer.

(a) Perform the initialization step and the first iteration of the basic genetic algorithm presented in Sec. 13.4 by hand. Follow the instructions given at the beginning of the Problems section to obtain the needed random numbers. Show your work, including the use of the random numbers.

(b) Use your IOR Tutorial to apply this algorithm. Observe the progress of the algorithm and record the number of times that a pair of parents give birth to a child whose fitness is better than for both parents. Also count the number of iterations where the best solution found is better than any previously found.

A 13.4-5. Follow the instructions of Prob. 13.4-4 for the nonconvex programming problem shown in Prob. 13.3-9 when the variable \(x\) is restricted to be an integer.

A 13.4-6. Follow the instructions of Prob. 13.4-4 for the nonconvex programming problem shown in Prob. 13.3-10 when both of the variables \(x_1\) and \(x_2\) are restricted to be integer.

A 13.4-7. Table 13.8 shows the application of the basic genetic algorithm described in Sec. 13.4 to the example of a traveling salesman problem depicted in Fig. 13.4 through the initialization step and first iteration of the algorithm.

(a) Use your IOR Tutorial to apply this same algorithm to this same example, starting from another randomly selected initial population and proceeding to the end of the algorithm. Does this application find the optimal solution \(1-3-5-7-6-4-2-1\) or, equivalently, \(1-2-4-6-7-5-3-1\)?

(b) Because of its use of random numbers, a genetic algorithm will provide slightly different results each time it is run. Use your IOR Tutorial to apply the basic genetic algorithm described in Sec. 13.4 to this same example five more times. How many times does it find the optimal solution?

13.4-8. Reconsider the traveling salesman problem shown in Prob. 13.1-1.

(a) Perform the initialization step and the first iteration of the basic genetic algorithm presented in Sec. 13.4 by hand. Follow the instructions given at the beginning of the Problems section to obtain the needed random numbers. Show your work, including the use of the random numbers.

(b) Use your IOR Tutorial to apply this algorithm. Observe the progress of the algorithm and record the number of times that a pair of parents gives birth to a child whose tour has a shorter distance than for both parents. Also count the number of iterations where the best solution found has a shorter distance than any previously found.

A 13.4-9. Follow the instructions of Prob. 13.4-8 for the traveling salesman problem described in Prob. 13.2-5.

A 13.4-10. Follow the instructions of Prob. 13.4-8 for the traveling salesman problem described in Prob. 13.2-6.

A 13.5-1. Use your IOR Tutorial to apply the basic algorithm for all three metaheuristics presented in this chapter to the traveling salesman problem described in Prob. 13.2-5. (Use \(1-2-3-4-5-6-7-8-1\) as the initial trial solution for the tabu search and simulated annealing algorithms.) Which metaheuristic happened to provide the best solution on this particular problem?

A 13.5-2. Use your IOR Tutorial to apply the basic algorithm for all three metaheuristics presented in this chapter to the traveling salesman problem described in Prob. 13.2-6. (Use \(1-2-3-4-5-6-7-8-9-10\) as the initial trial solution for the tabu search and simulated annealing algorithms.) Which metaheuristic happened to provide the best solution on this particular problem?