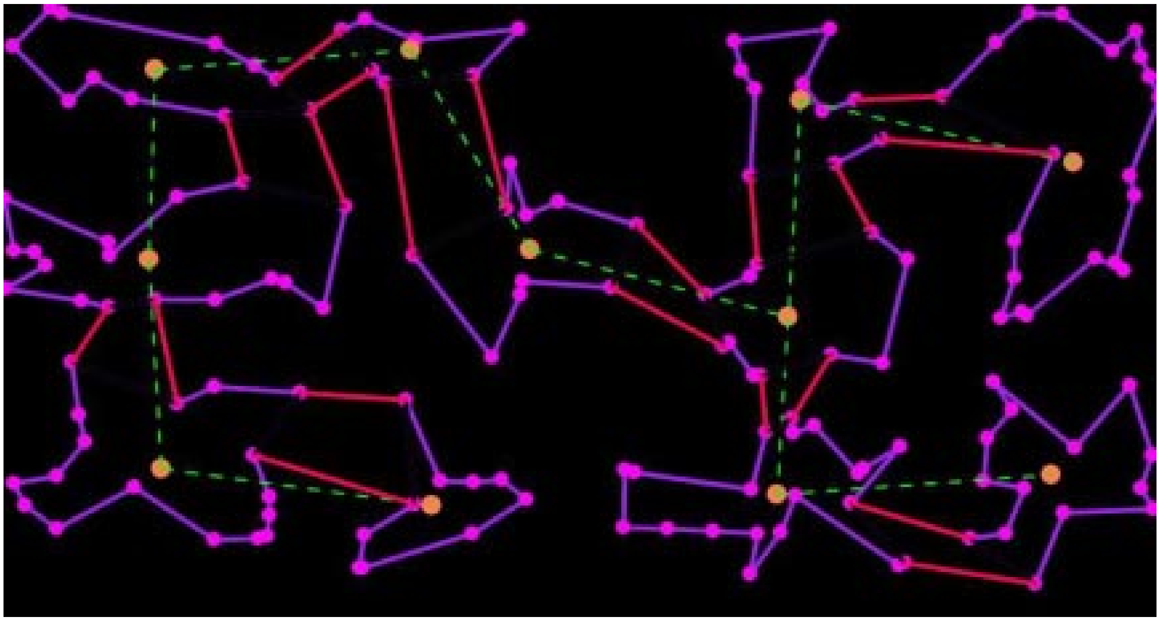
Selim Tezel
Master's Project
Advisor: Pascal Van Hentenryck
Spring 2002

Traveling Salesman's SketchPad (TSSP) 1.0 Beta

Tutorial



Traveling Salesman's SketchPad (TSSP) 1.0 Beta Tutorial

Let us perform some tasks in TSSP to discover some of its features and capabilities

(A) Basics:

Task: Construct a Node

1)Click on the Node Tool of the ToolBox

2)Notice the cursor changes when mouse moves over to sketch now

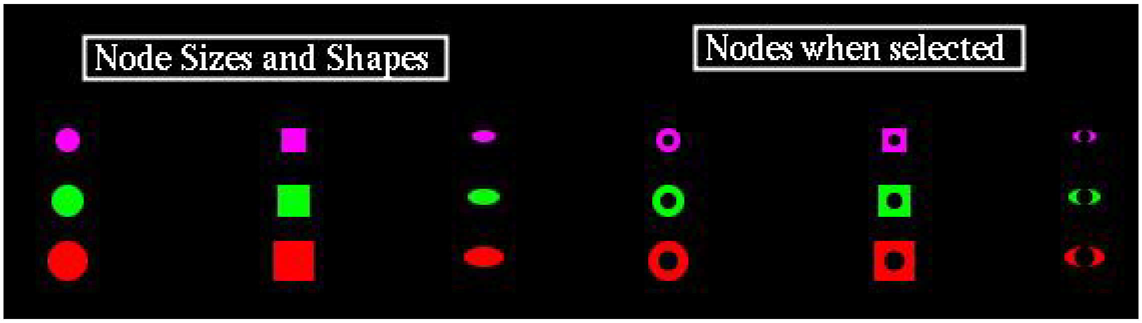
3)Click on a blank area of the drawing panel

4)Notice a Node has been constructed at the location specified by the click

5)Click on the Selection Tool of the ToolBox to end Node construction

COMMON MISTAKE:
Trying to drag a Node after its construction with the Node Tool. Nodes can only be dragged by the Selection Arrow, so always return to Select Arrow when done with a free-hand Tool.

Task: Change the attributes of a Node



1)Click the Select Arrow on a blank area and then on the Node

2)Notice how a black dot (indicating selection) appears on Node when clicked on. This is how we select Nodes. To make multiple selections press the <SHIFT> key on your keyboard while doing the selections.

3)Location of a free Node such as the one we just constructed can be changed by dragging the Node with the mouse. To fix the location of a Node to make it undraggable, select the Node and go to
<Transform> Menu <Set Node...Fixed> option.
To make it drag-able again, use the
<Transform> Menu <Set Node...Unfixed> option.

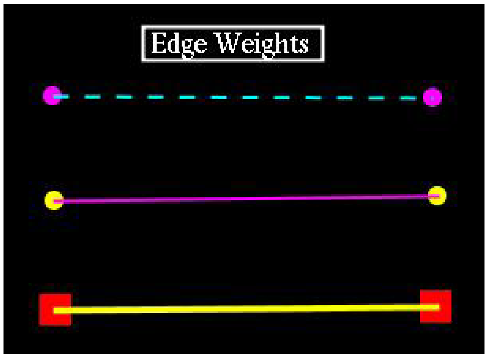
4)To change the color of a Node, first select it then go to
<Display> Menu <Color> options
and choose a color

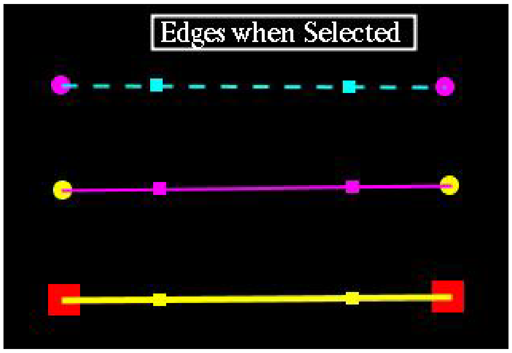
5)Similarly, size and shape of a Node can be changed via the
<Display> Menu <Node Size>, <Node Shape> options

6) A Node can be hidden, once it is selected via
<Display> Menu <Hide Objects> option.
To make it visible again go to
<Display> Menu <Show All Hidden> option.

7)A Node can be permanently deleted, once it is selected by either hitting delete key on the keyboard or via the
<Edit> Menu <Clear> option

Task: Construct an Edge





1)Click on the Edge Tool of the Tool Box

2)Click, drag and release the mouse in the drawing panel to construct an edge.

3)Click back on the Select Arrow. Click on the Edge and then on a blank area. Notice how Edge selection is indicated.

4)Select and move around the end Nodes of the Edge to extend the Edge.

5)Select the Edge and change the Edge attributes (color and weight) via the <Display> menu <Color> <Edge Weight> options.

6)Notice that the Edge can only be dragged by dragging both of its end Nodes.

Questions:

How else can an Edge be constructed? (Hint:<Construct> Menu)

What happens if your answer to above is applied when multiple Nodes have been selected?

What happens when you start your free-hand Edge constructor on an existing Node? or end on an existing Node?

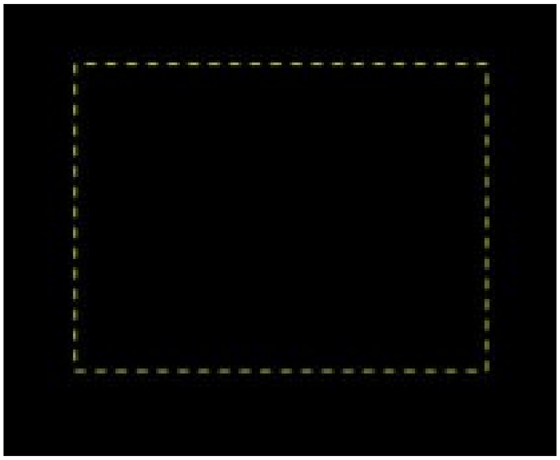
How can you hide an edge?

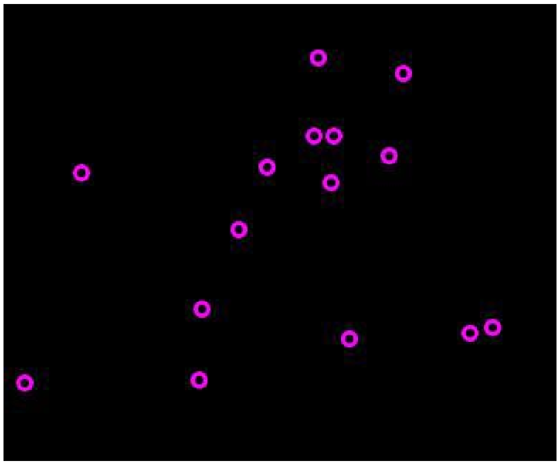
How can you make an Edge unmovable?

How can you delete an Edge?

What happens to an Edge when one of the end Nodes is deleted?

Task: Construct 100 Random Nodes





1)Use Marquee Tool to define a rectangular region (Click Selection arrow on blank area, drag and release when you are satisfied with the size of the rectangular area)

2)Go to
<Construct> Menu <Random Nodes> option.

3)A dialog will appear, enter 100 for the number of Nodes, and hit OK.

4)By default created objects will be set in selected mode immediately after their creation (and will remain so until blank area is clicked by the user with the Select Arrow) Change the color, size and shape of the 100 Nodes all at once by going to the <Display> Menu options.

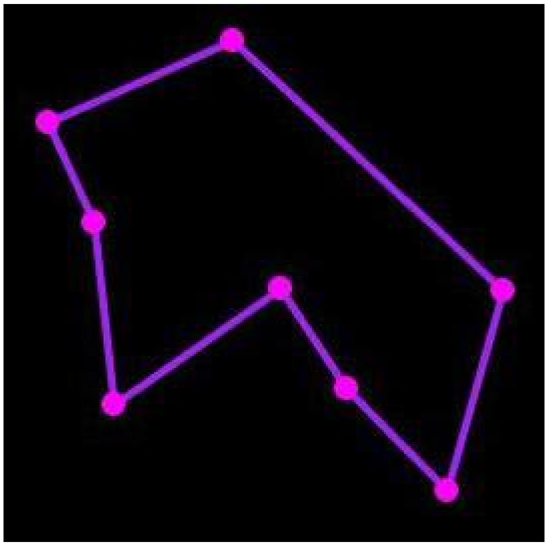
5)Once the Nodes are all selected, you can move them all by dragging any one of them.

Question:
How can you construct 1000 Random Nodes in a few seconds?

How can you fill the Drawing Panel non-uniformly with random Nodes?

(B) Intermediate

Task: Construct a Tour Through Eight Randomly located Nodes.



1)Construct the 8 random Nodes

2)Making sure that all Nodes are selected, go to
<Construct> Menu <Shortest Tour> option
and select an algorithm such as <Farthest Insertion Algorithm>

3)Notice the constructed Tour is in selected mode following its creation

4)While the Tour is selected change the edge weight and color of the Tour via <Display> Menu options.

5)Drag one of the Nodes of the Tour, notice how the Tour regenerates itself each time.

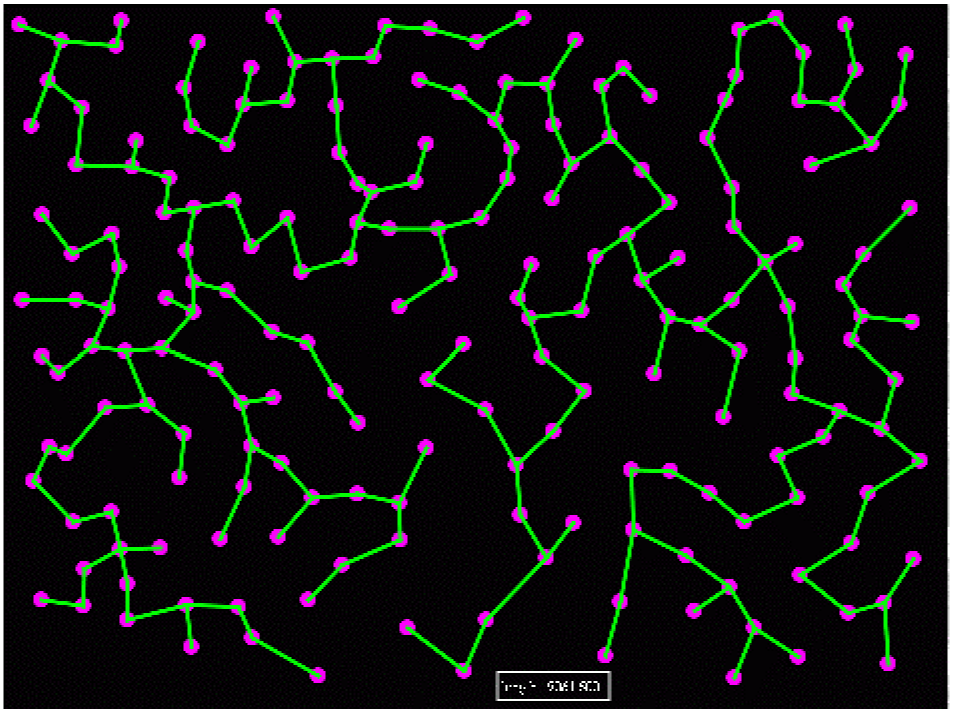
6)Notice the Tour edges cannot be dragged. To drag the Tour all the parent Nodes must be dragged.

7)Select the Tour and measure its length via
<Measure> Menu <Length> option.

8)Notice how dragging a Node, updates the length measurement

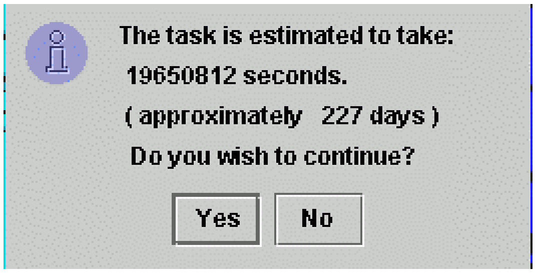
Questions:

-How do you construct a Minimum Spanning Tree?



-How can you construct a Path with a set Node as the Start of Path?

-Which Tour algorithm though yields the best results is the slowest?

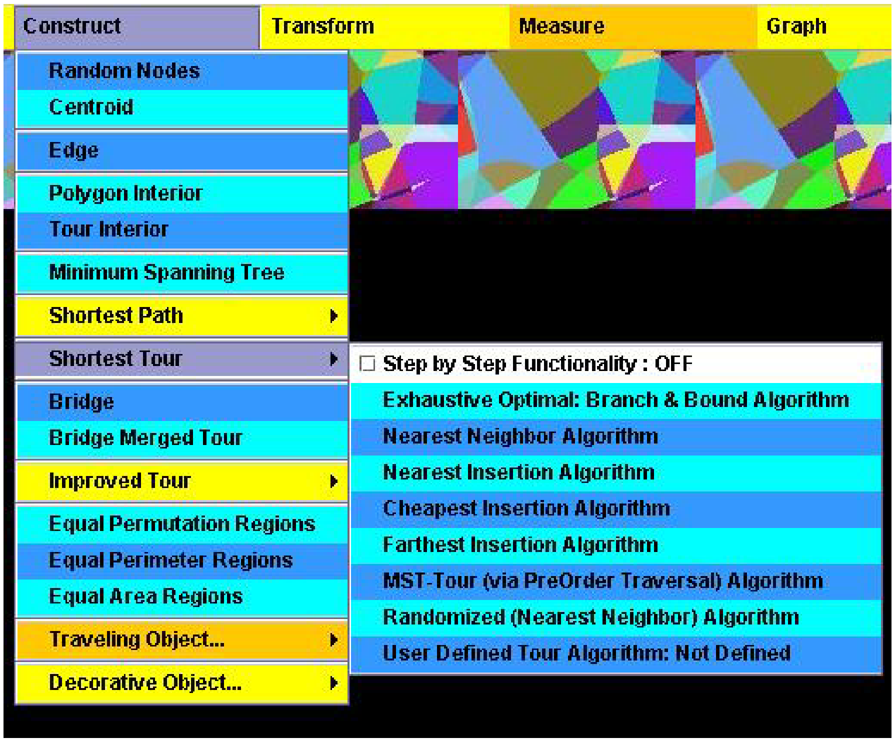


The task is estimated to take:
19650812 seconds.
(approximately 227 days)
Do you wish to continue?

Are there any algorithms yielding self-crossing Tours?

-How can you find the area enclosed by a Tour? (Hint: Construct something else prior to measurement)

Task: Generate a Tour step-by-step


Construct
Transform
Measure
Graph
Random Nodes
Centroid
Edge
Polygon Interior
Tour Interior
Minimum Spanning Tree
Shortest Path
Shortest Tour
Bridge
Bridge Merged Tour
Improved Tour
Equal Permutation Regions
Equal Perimeter Regions
Equal Area Regions
Traveling Object...
Decorative Object...
□ Step by Step Functionality : OFF
Exhaustive Optimal: Branch & Bound Algorithm
Nearest Neighbor Algorithm
Nearest Insertion Algorithm
Cheapest Insertion Algorithm
Farthest Insertion Algorithm
MST-Tour (via PreOrder Traversal) Algorithm
Randomized (Nearest Neighbor) Algorithm
User Defined Tour Algorithm: Not Defined

1)Construct 50 random Nodes, and while they are selected

2)Go to
<Construct> Menu <Shortest Tour> option.

3) Turn <Step by Step Functionality> On by choosing it.

4)Repeat step 2, notice that now <Step by Step Functionality> has been turned On.

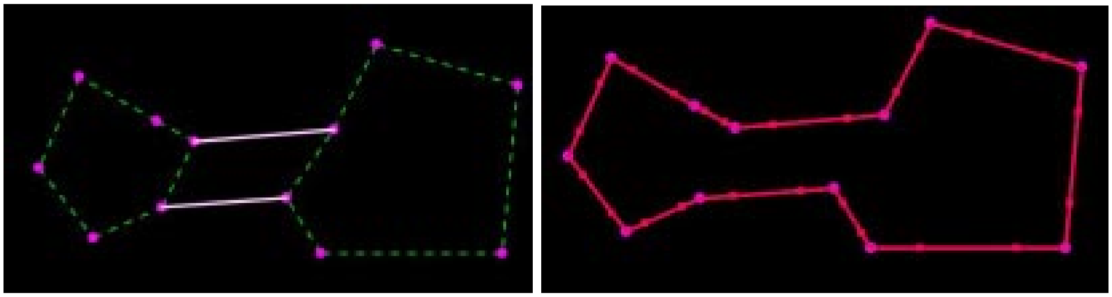
5)Now choose an algorithm, such as <Nearest Neighbor Algorithm>

6)A dialog allowing stepping backwards and forwards will appear, play with it.

Questions:
How do you turn the stepping off?

Why do some algorithms yield self-crossing tours?

Task: Building a Bridge Merged Tour



1)Construct two separate Tours

2)Select both Tours (remember using the <SHIFT> key for multiple selections)

3)Go to
<Construct> Menu <Cheapest Bridge> option
to create a Bridge

4)While the Bridge is selected go to
<Construct> Menu <Bridge Merged Tour> option

5)A dialog will ask you if you wish to hide the original tours and the bridge, click Yes.

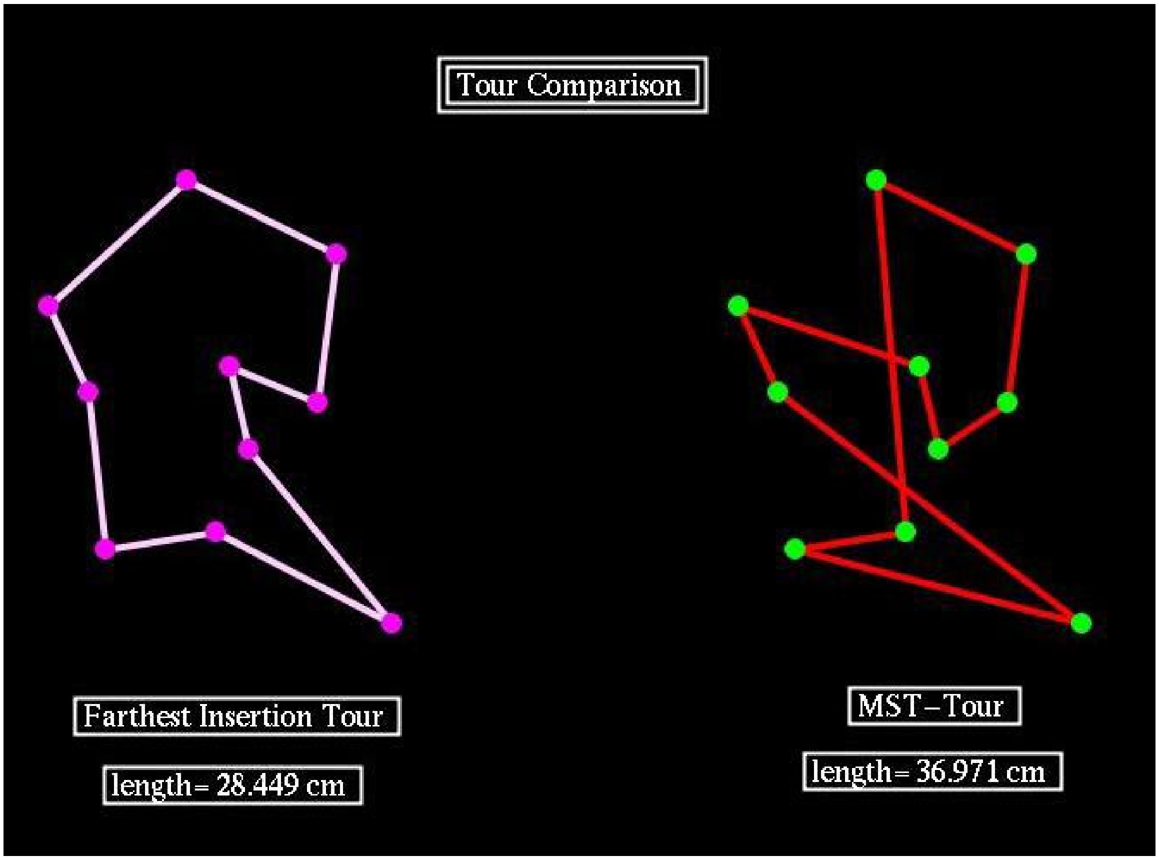
6)Notice how the Merged Tour updates itself as you move some of its Nodes

Questions:
How can you make visible the original Tours? Hide them back again?

How can you measure the length and area of the Merged Tour?

(C) Advanced:

Task: Tour comparison



1)Near the left side of the Drawing Panel (close to the Tool Box) construct ten Nodes

2)Select these Nodes and go to
<Transform> Menu <Translate> option

3)A dialog will appear asking for the x and y translations. Enter, say, x=400, y=0 for translations

4)Color the new Nodes with a color different than the originals

5)Move some Nodes in the first group and notice how they affect the second group

6)Notice that the second group is dependent on the first and cannot be dragged by themselves

7)Select the first group, construct a Tour through them via the <Construct> Menu <Shortest Tour...Farthest Insertion Algorithm> option.

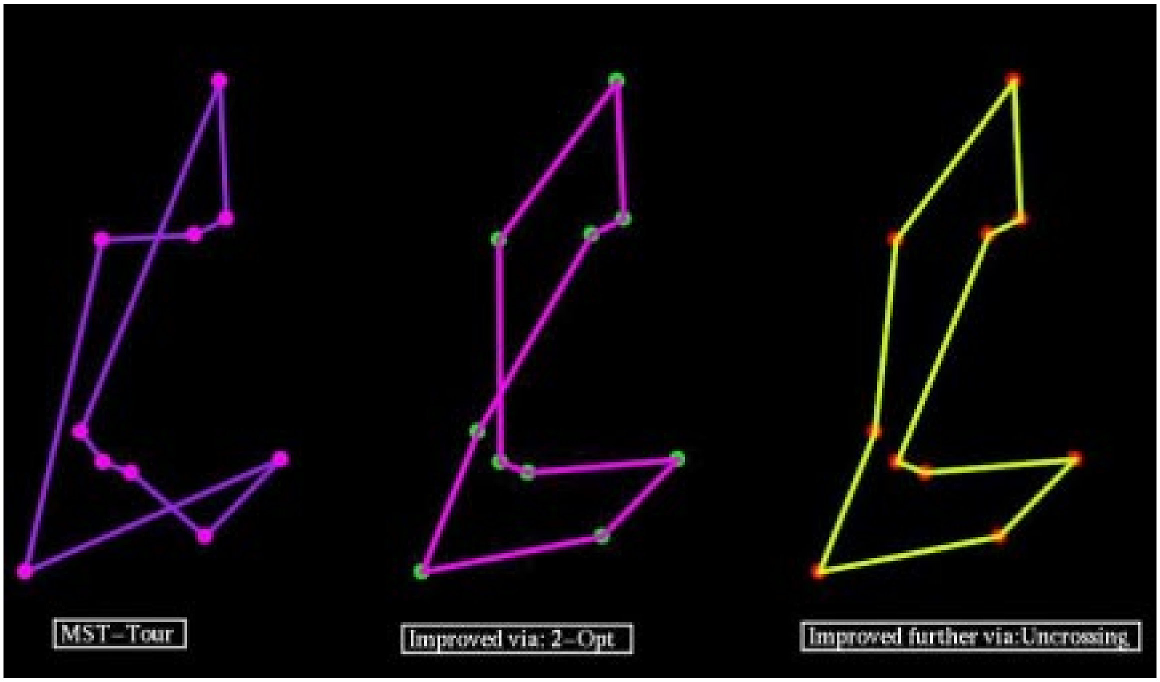
8)Now select the second group, construct a Tour through them via the <Construct> Menu
<Shortest Tour...MST-Tour (via Pre-Order Traversal) Algorithm> option.

9)Move some Nodes of the first group, to see how both Tours compare.

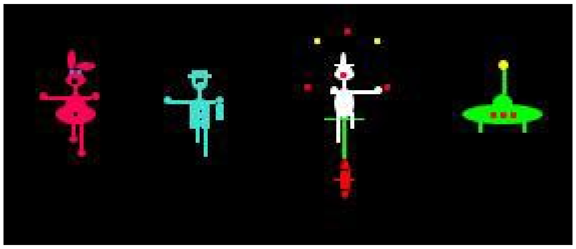
10)Compare Tour lengths by taking measurements via <Measure> Menu <Length> option.

Questions:
The exhaustive optimal always yields the best Tour but takes too much time, which other Tour algorithm is superior to others in terms of yielding good results quickly? Which is the worst?

Can a bad Tour be improved in TSSP?(Hint:<Construct> Menu option...)



Task: Generating an animated traveler



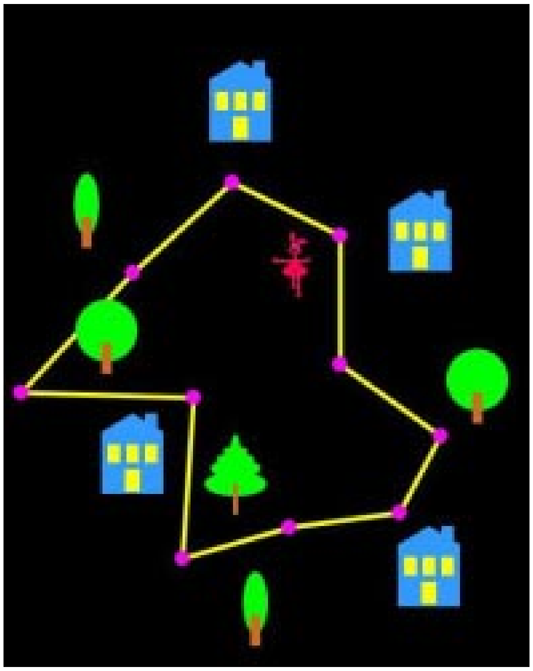
1) Generate a Tour

2)While the Tour is selected go to
<Construct> Menu <Traveling Object> Menu

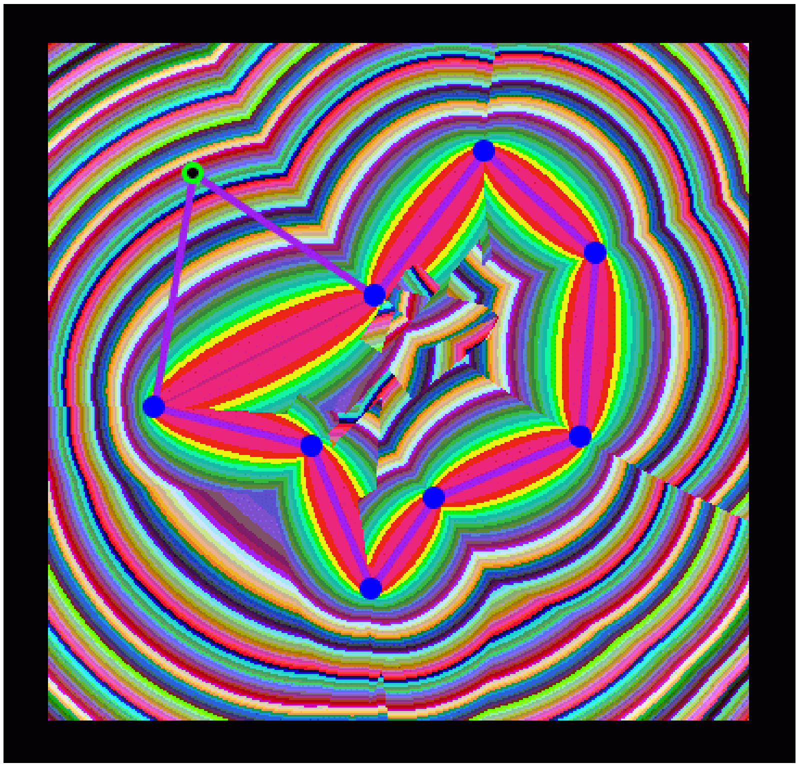
3)Choose a traveler according that fits your fancy

Questions:
How do you re-animate a traveler once it stops traveling? (Hint:<Display> Menu)

How do you create some Decorative Objects, such as a House, Vegetation in your sketch?



Task: Construct Equal Perimeter Regions



0)Take a deep breath

1)Construct a Tour (say from 10 Nodes via Farthest Insertion Algorithm)

2)Choose a Node (it will your 'probe Node') and color it, say, green, to look different than the others

3)Select this Node and set it to be the probe via
<Transform> Menu <Set Node...Isomorphism Probe> option

4)Use the Marquee Tool to select the Tour and all its Nodes while defining a rectangular region

5)Go to
<Construct> Menu select <Equal Perimeter Regions> option

6)A dialog will appear asking for a resolution value, enter 20 and then a bandwidth value, enter 20. (It is recommended that you always enter a large resolution value initially to get a quick initial picture.)

7)Notice the emerging shape, move your probe around, then move another Node, notice what happens

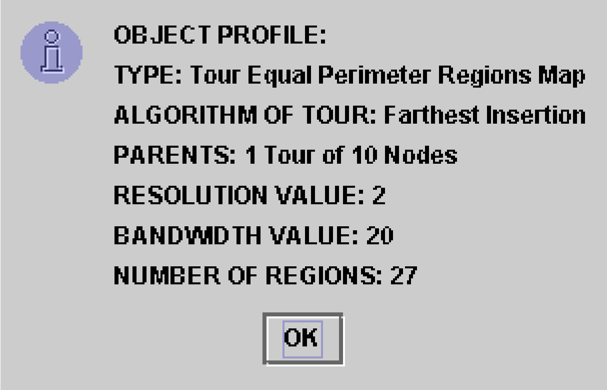
8) Select the Region by clicking on a part of it

9)Go to
<Display> Menu <Reset Isomorphism Bandwidth> option,
put a new value, say 10 , notice what changes

10)Go to
<Display> Menu <Reset Isomorphism Resolution> option,
put a small value, notice what happens.

11)Select the Tour (you will need to press <SHIFT> key to deselect the Region first) and measure its length. Move the probe around and notice how within a colored region, the length stays within the bandwidth range.

Questions:
What tool in the ToolBox will give you information about the resolution or bandwidth values of an already created Isomorphism Region?



OBJECT PROFILE:
TYPE: Tour Equal Perimeter Regions Map
ALGORITHM OF TOUR: Farthest Insertion
PARENTS: 1 Tour of 10 Nodes
RESOLUTION VALUE: 2
BANDWIDTH VALUE: 20
NUMBER OF REGIONS: 27

How do you construct Equal Permutation Regions?

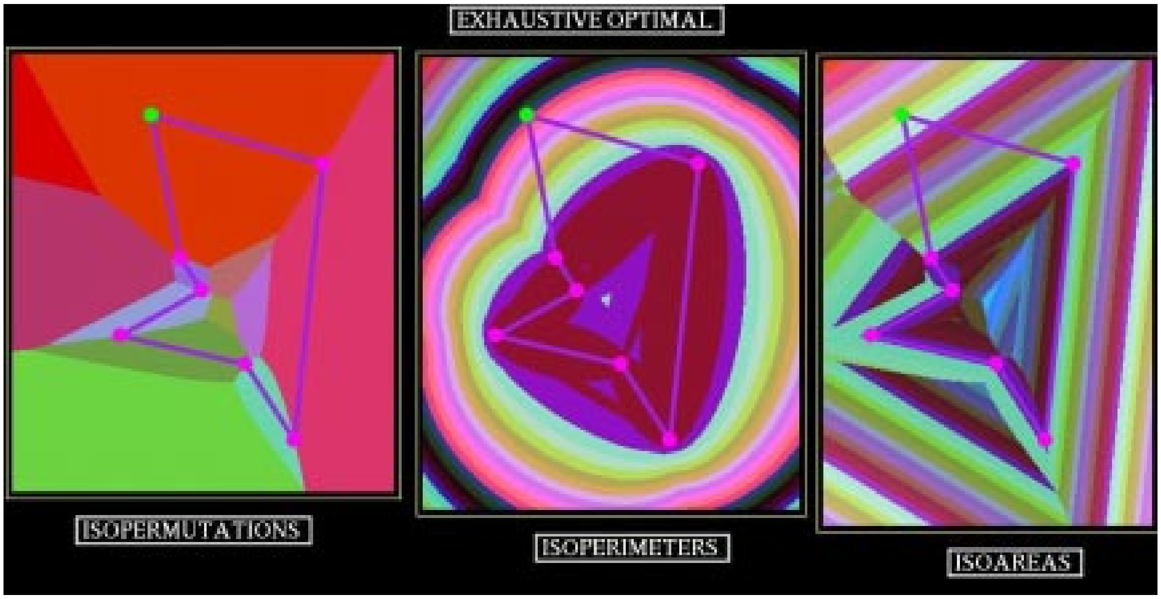
What is the function of the probe Node?

Why is it advisable to fix non-probe Nodes of Isomorphism Regions at best resolution values?

What does resolution value, bandwidth measure?

Where does the funky image on the TSSP frame border come from?
(Hint:<Help> Menu <About TSSP>)

Can you build a presentation sketch that displays the three types of Isomorphism Regions for the same Tour, side by side?
(Hint: <Transform> Menu <Translate> option.)



What is the meaning behind these images? Do they demonstrate anything of use or importance?

Task: Writing and Loading a TSSP data File

1)Using you favorite text editor, create a simple text file that contains the following data regarding the x and y location of Nodes to be constructed:

```
100,200
100,400
300,400
300,200
```

Make sure the file starts and ends with data and doesn't start or end with any blank lines.(Otherwise file will not load)

2)Give the file a name such as "mydatafile.nodes" and save it.

3)Go to
<File> Menu <Load Data File> option

4)Find and load the data file you just created

5)Notice the Nodes created via data loading cannot be dragged (why?)

Questions:
How can you make the Nodes created via data loading, draggable?
(Hint:<Transform> Menu <Set Node....> option)

What happens if the same data file were named with name such as: "mydatafile.tour" or "mydatafile.path"? (These are the only other acceptable file name endings)

What happens, if your data file contains Node locations beyond the boundary of the Drawing Panel?
(Hint: You have two options. One will change the data, the other will change the Drawing Panel Scale)

(D) Most Advanced:

Task: Writing your own Tour Algorithm

In TSSP you can write and visualize your own Tour algorithms. The code will be in Java and once compiled should be incorporated into the regular tools of the program.

1)Find the file named:
"TSSPUserDefinedTourAlgorithm.java" in the TSSP directory

Unedited it looks like:

```
package TSSP;
//this import will allow you to use short name Vector for java.util.Vector
import java.util.*;

public class TSSPUserDefinedTourAlgorithm implements TSSPTourAlgorithm
{
  private String _nameOfTourAlgo;

  public TSSPUserDefinedTourAlgorithm()
    {
      //replace "Not Defined" with a string that will be
      //displayed in the menu options as the name of your tour algo

      _nameOfTourAlgo="Not Defined";//change this string
    }
  public String nameOfTourAlgo()
    {
      return _nameOfTourAlgo;
    }
```

```
 public Vector findBestTour(Vector initialGuess,
                                        Vector nodesList)
    {
      //fill this method: given a vector of nodes
      //return a vector of nodes IN ORDER indicating the tour

      //don't include the first node again at the end of the
      //vector, the tour class will connect last to first
      //automatically when drawing the tour

      //if you want you can totally ignore the user input
      //initialGuess which is there to save computation time
      //for some algos such as the exhaustive algo
      //if you use it, just make sure you consider the cases
      //when it is null, ie user did not make an initial guess

      return null;//return here a Vector containing the nodes in tour order

    }


  public Vector generateSteps(Vector initalGuess,Vector nodesList)
    {
        //if you wish to define steps for your algo fill this in, else leave it as is.
        //the idea is that algo could store intermediate states and return a vector of vectors
        //(of TSSPNodes) that could be displayed as steps of the algo
        //if this makes no sense, best to leave this method as is.

      return null;
    }

  public boolean isDefined()
    {
      //once you have given a name to the algo and
      //written the findBestTour method
      //have this method return true

      return false; //as a last step replace false with true

    }

}
```

2)Edit the file, save and compile:
javac TSSPUserDefinedTourAlgorithm.java

to do something useful. A basic example is given below:

```
package TSSP;

//this import will allow you to use short name Vector for java.util.Vector
import java.util.*;

public class TSSPUserDefinedTourAlgorithm implements TSSPTourAlgorithm
{
  private String _nameOfTourAlgo;

  public TSSPUserDefinedTourAlgorithm()
    {
     //replace "Not Defined" with a string that will be
     //displayed in the menu options as the name of your tour algo

     _nameOfTourAlgo=" MyVeryOwnTourAlgo";//changed this string

    }


  public String nameOfTourAlgo()
    {

     return _nameOfTourAlgo;
    }

  public Vector findBestTour(Vector initialGuess,
                                   Vector nodesList)
    {
          //here is the added code
          //here is a very simple Tour algo:  if the user had an initial guess for the order of
          //the tour, that  order becomes the tour, otherwise the order in which nodes have
          //been selected by the user becomes the tour.

          if(initialGuess!=null)
               return initialGuess;
         else
                return nodesList
    }
  public Vector generateSteps(Vector initalGuess,Vector nodesList)
    {
        //if you wish to define steps for your algo fill this in, else leave it as is.
       //the idea is that algo could store intermediate states and return a vector of vectors
       //(of TSSPNodes) that could be displayed as steps of the algo
       //if this makes no sense, best to leave this method as is.


     return null; //recommend that you don't touch this quite yet
    }

  public boolean isDefined()
    {
     //once you have given a name to the algo and
     //written the findBestTour method
     //have this method return true

     return true; //as a last step replaced false with true

    }
}
```

Note:
In general you will need to do something clever with the parameter information passed in the method
findBestTour(Vector initialGuess, Vector nodesList)

(You can safely ignore the first parameter for most purposes)

It is recommended that you review Java API for methods associated with the very useful class java.util.Vector located at:

http://java.sun.com/products/jdk/1.3/docs/api/index.html

You will also need to know that TSSPNode class have the following methods:

```
public Point2D.Double   getLocation();
public double getX();
public double getY();
```

so a common useful thing to do may be :

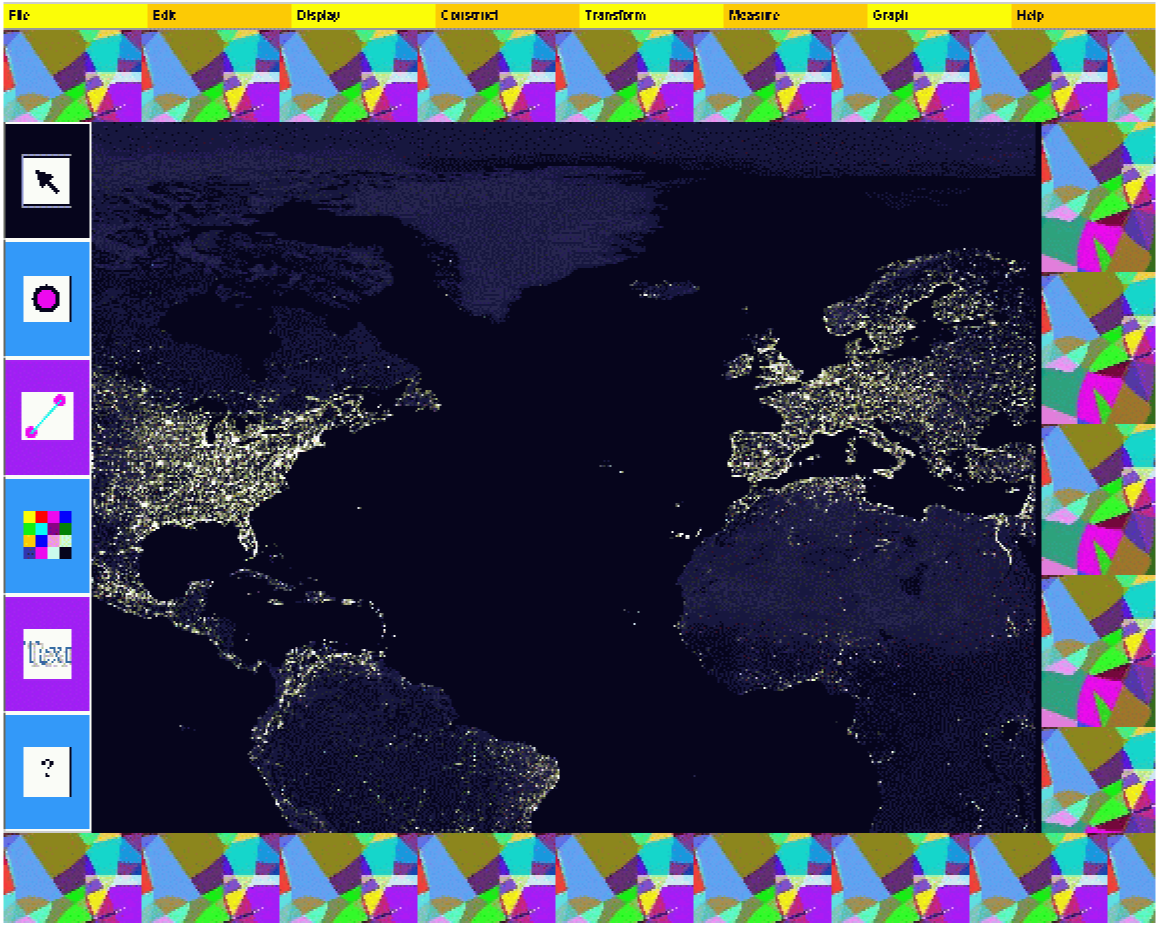
```
for(int i=0; i<nodesList.size(); i++)
{
 //we cast  as we extract the nodes
TSSPNode  node=(TSSPNode)nodesList.elementAt(i}
//extract from the nodes their location info
double x=node.getX();
double  y=node.getY();
//do some useful calculations here.........

}
```

The ability to incorporate your own algorithms to TSSP is probably one of the most important and rewarding features of the software.   It is the hope of the author that the users will take advantage of this.

Miscellaneous Questions:

How do you import a background image to the sketch such as the map of USA? (Hint <File> Menu)

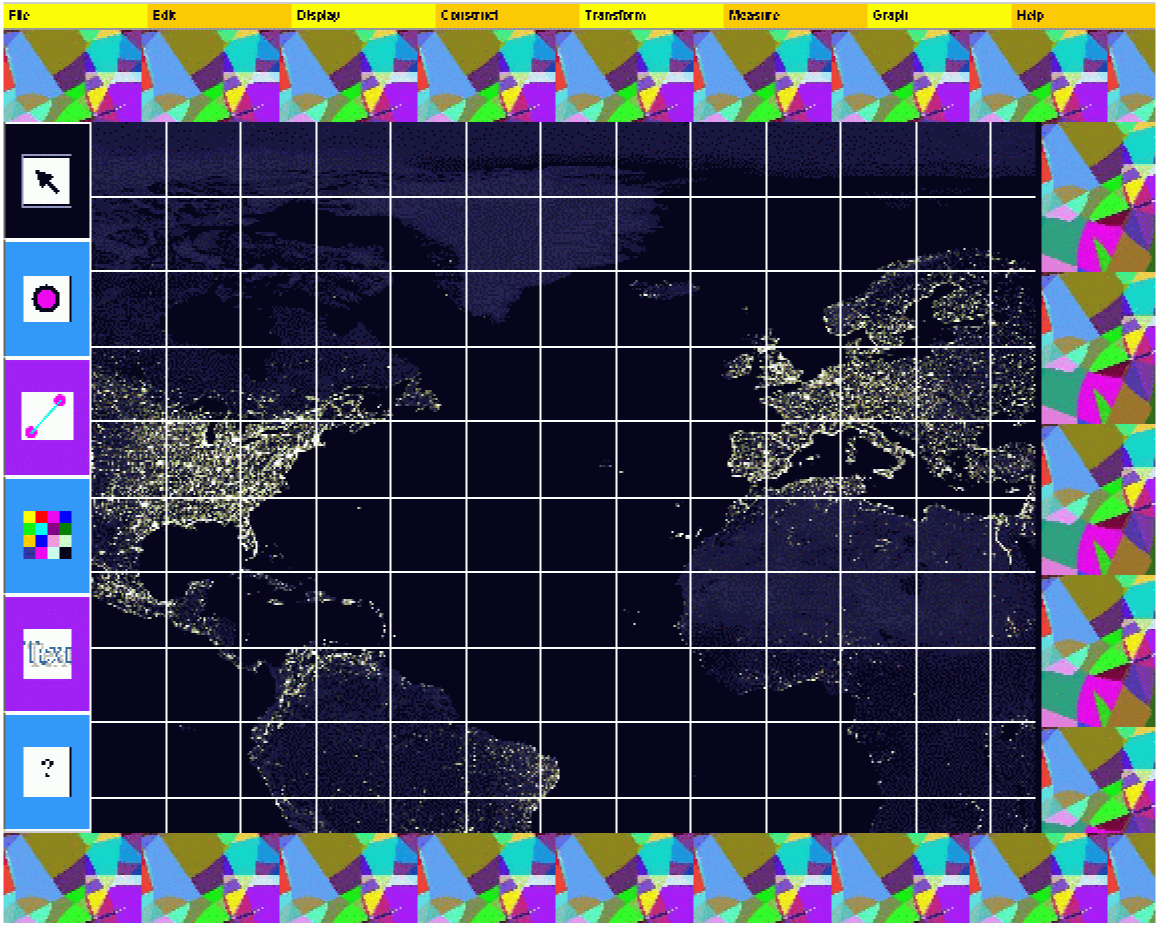


How do you delete an imported image? (Hint: <Edit> Menu)

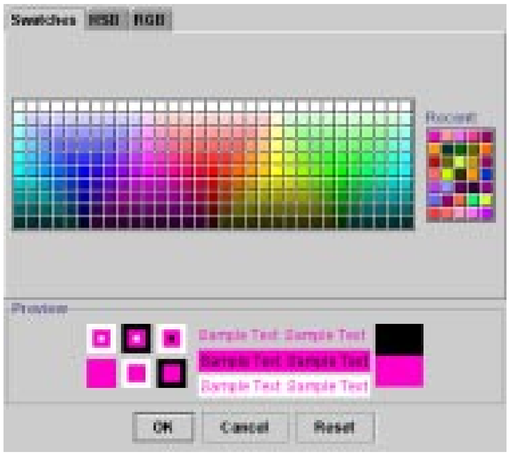
You imported a map and discovered that the map scale indicates that every 100 pixels on the drawing panel correspond to 1875 miles in the real world. How do you inform TSSP of this, so that your measurements reflect this real world scale?(Hint: <Graph> Menu <Set...>)

Can you save your work in any way in TSSP 1.0? (Hint: <File> Menu)

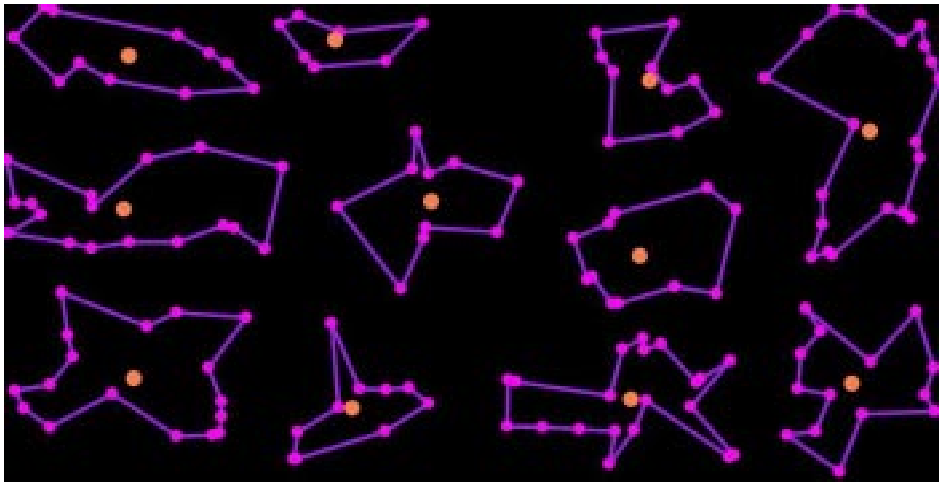
How do you turn on/off a grid coordinate system?(Hint: <Graph> Menu)

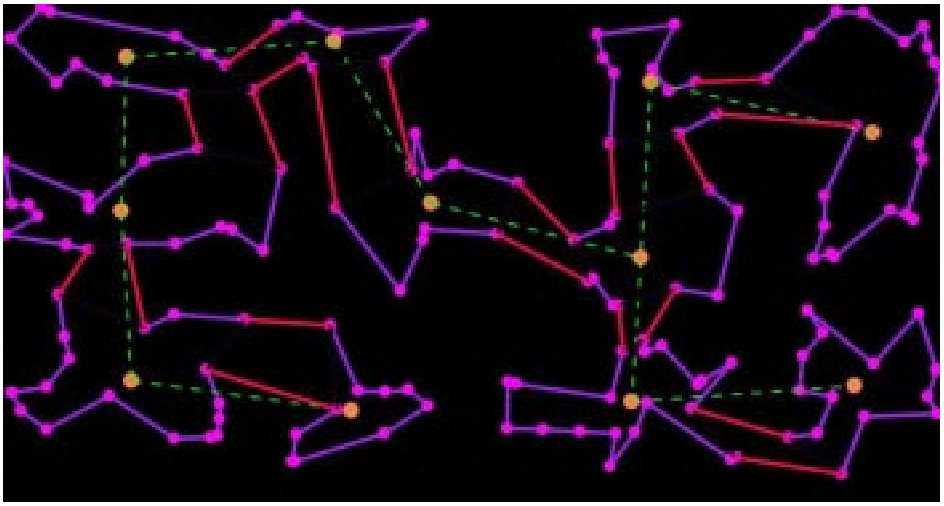


How can you access a richer coloring tool than that available at the <Display> menu?

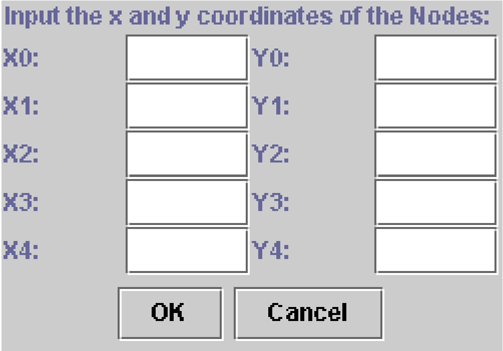


How can the <Construct> Menu <Centroid> option be used to devise a "Divide Conquer" approach to approximately solve a shortest Tour problem involving, say, 500 Nodes?





Can you plot a Node exactly at location (300,550)? (Hint: <Graph> Menu)



Input the x and y coordinates of the Nodes:

X0:		Y0:	
X1:		Y1:	
X2:		Y2:	
X3:		Y3:	
X4:		Y4:	

Where is the TSSP 1.0 coordinate system origin? (Hint: <Help> Menu)

How can you re-title a measurement?(Hint: Text Tool of the ToolBox)

The measurements keep appearing on top of another and become unreadable, what to do? (Hint: Measurements are objects)

What is an Improved Tour? (Hint: <Help> Menu)

Where can you find details of the algorithms used in the TSSP?
(Hint: <Help> Menu)

What software inspired the creation of TSSP? (Hint: <Help> Menu)

How can you contact the author of TSSP? (Hint: <Help> Menu)