Route planning

Route planning – Path planning

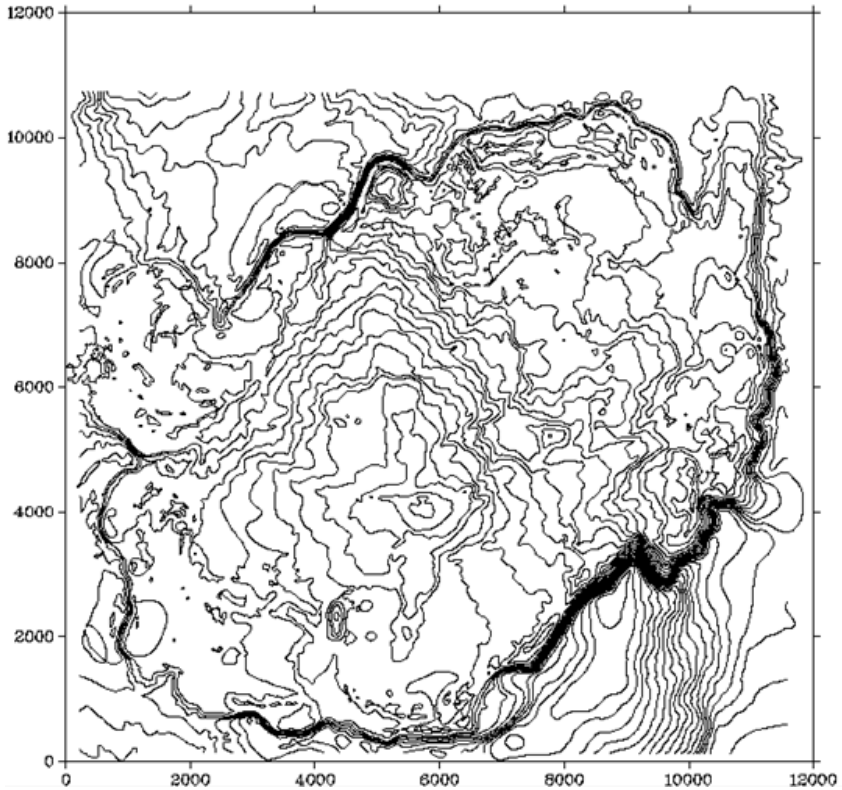
Route planning:
Is the process of finding the most efficient way to get from Point A to Point B.

Path planning:
A collision-free path from the start to the target according to an evaluation standard in the obstacle environment.
Find the shortest and most obstacle-free path from a start to goal state for an autonomous vehicle or a robot.

Route planning

Contour Map

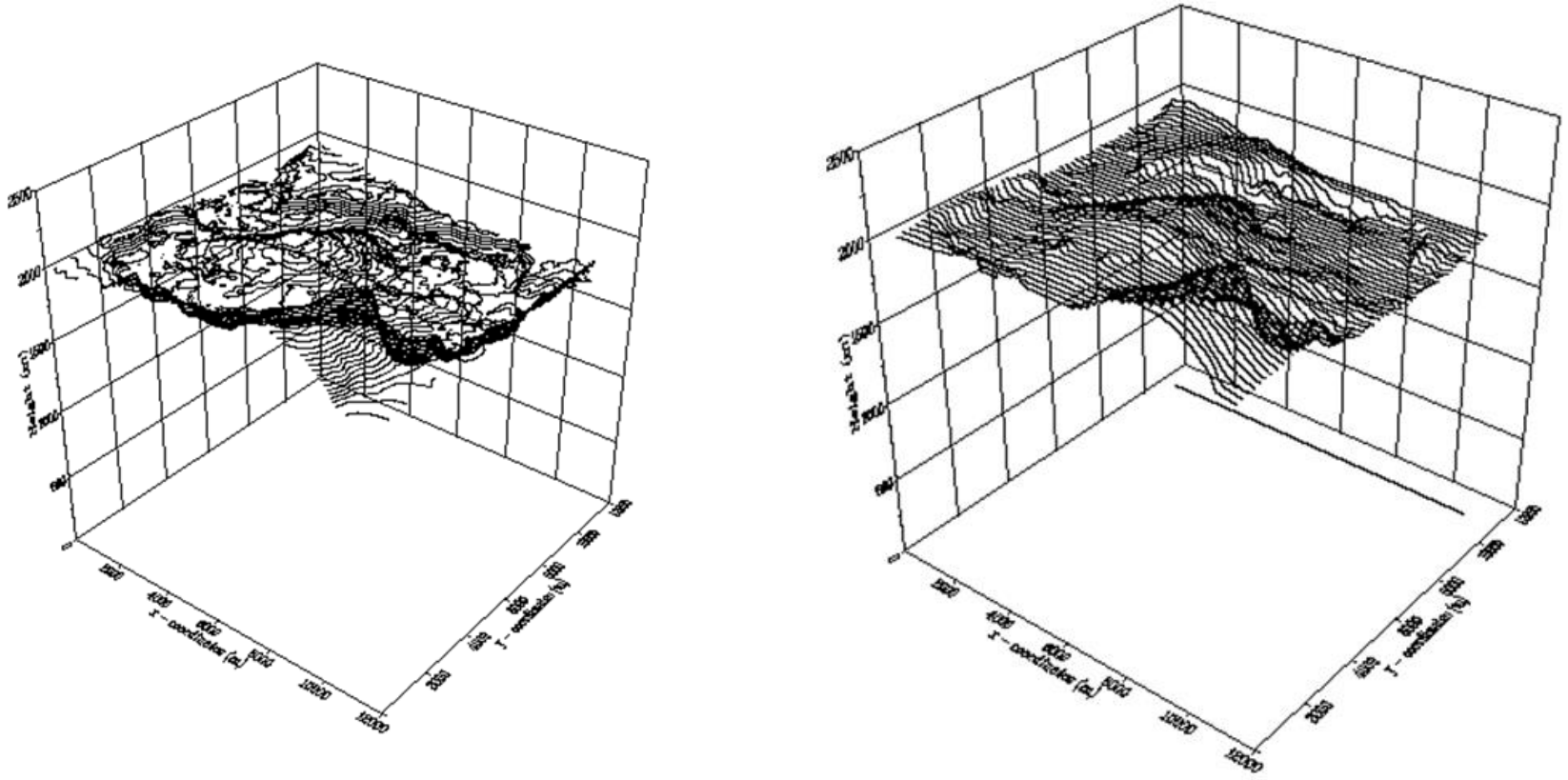
Menengai geothermal field



Route planning

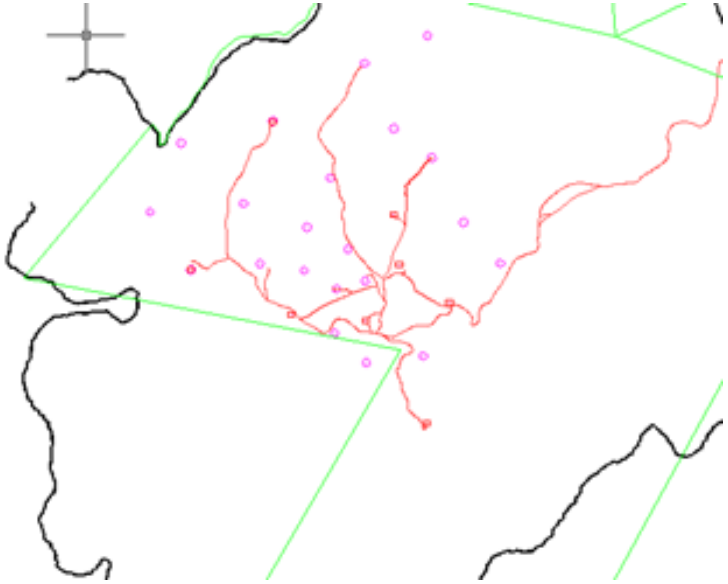
Digital elevation model (DEM):

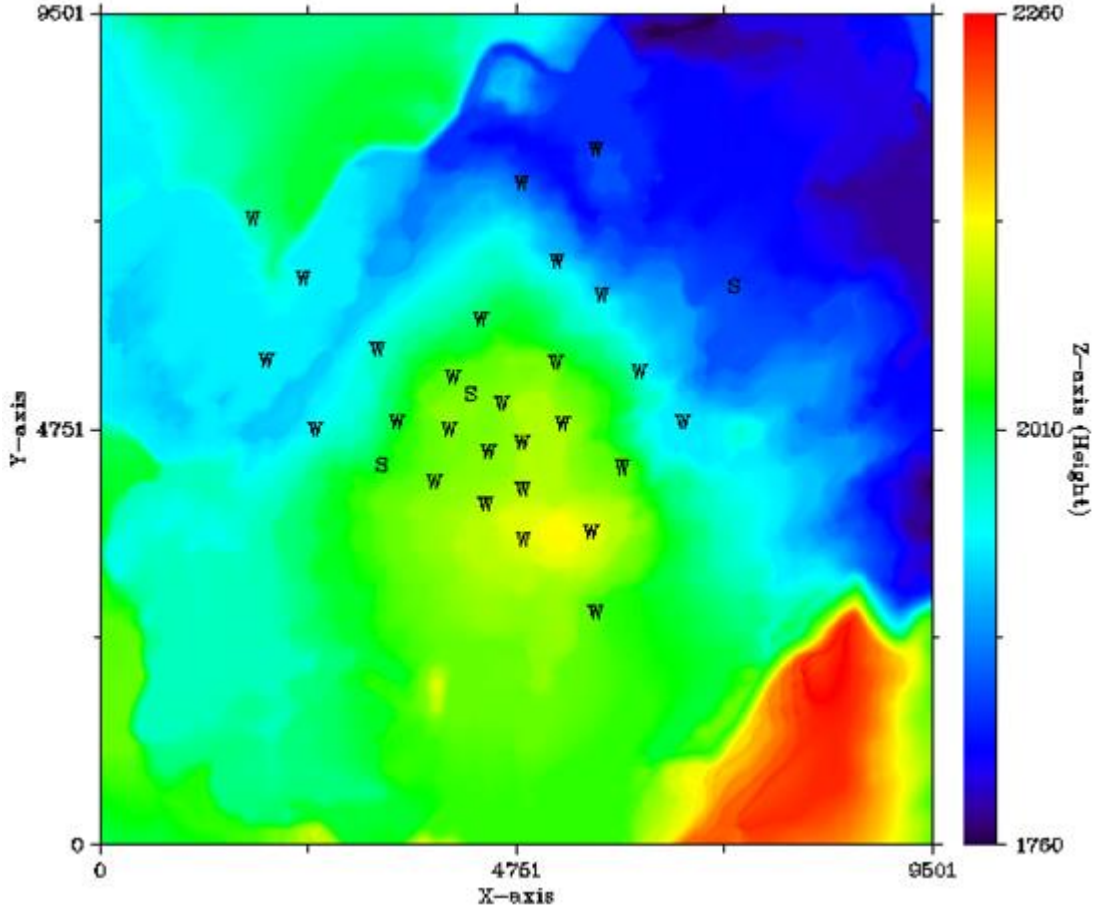
Digital representation of a given ground topography.



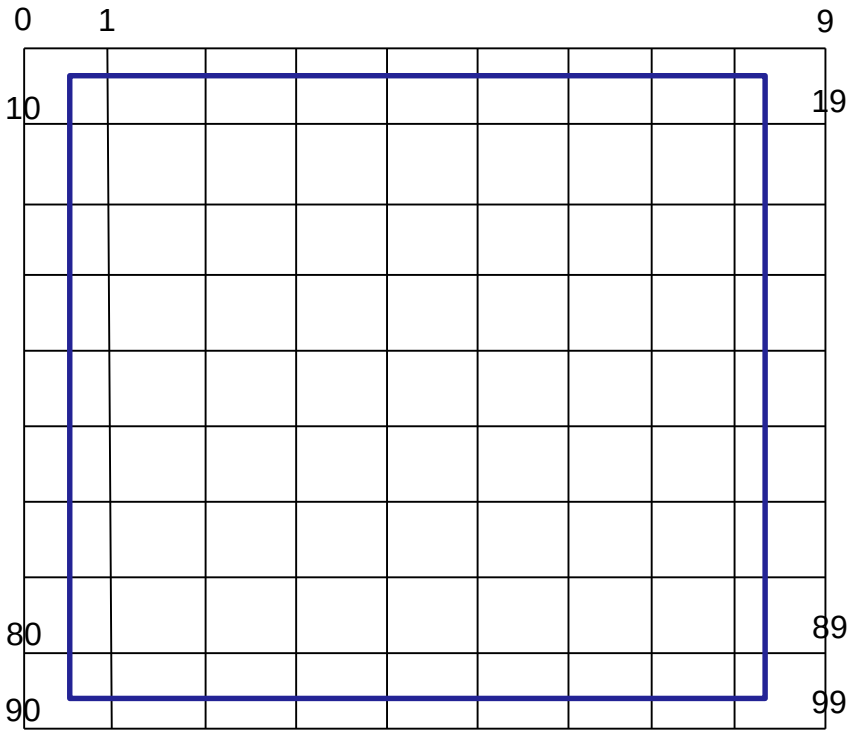
Digital Elevation Model

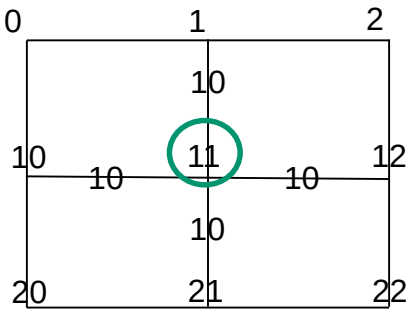
DEM Resolution: 1 m



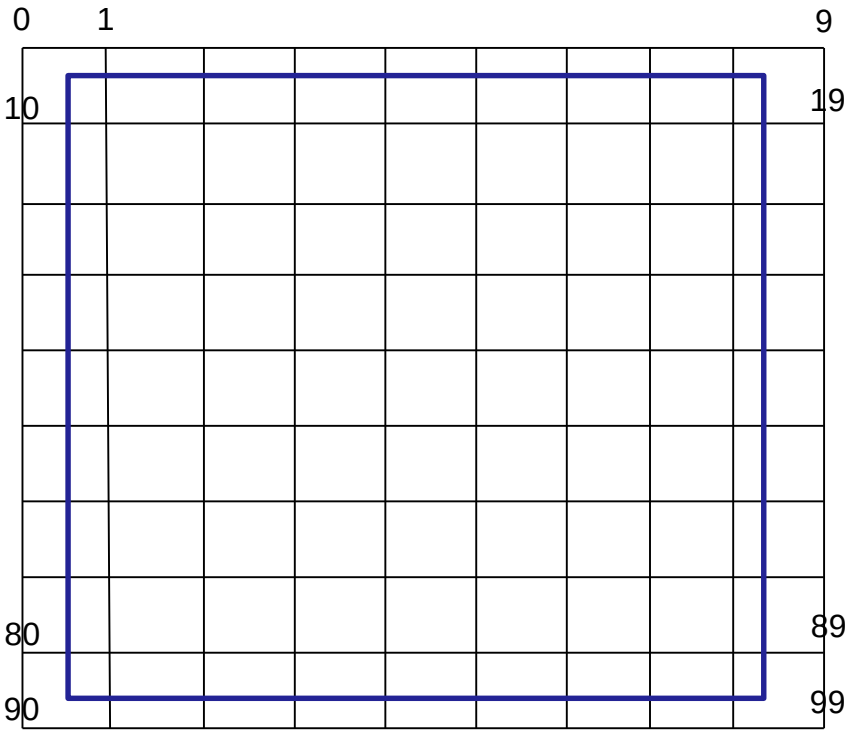


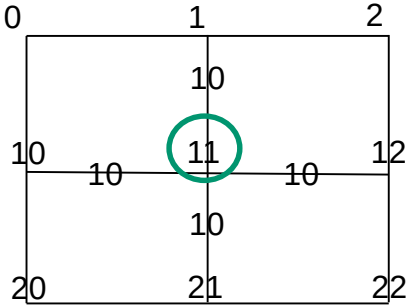
Example Lecture Assignment #16





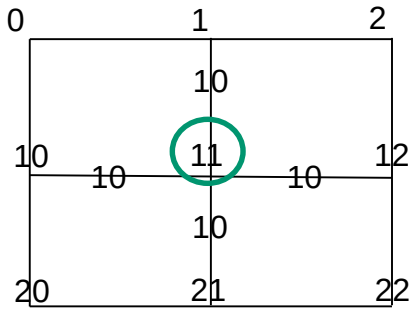
Example Lecture Assignment #16





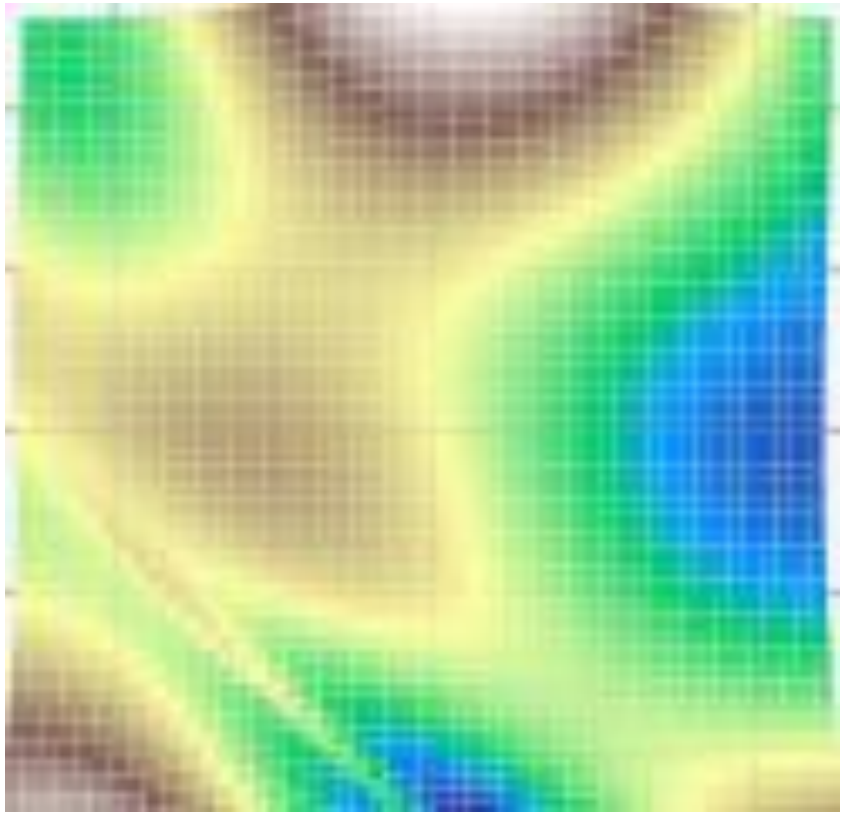

```
for i in range(8):
    for j in range(8):
        ip = (i+1)*10+(j+1)
        R.matrix[ip,ip-10] = 10.0
        R.matrix[ip,ip-1] = 10.0
        R.matrix[ip,ip+1] = 10.0
        R.matrix[ip,ip+10] = 10.0
```

Example Lecture Assignment #16




```
for i in range(8):
    for j in range(8):
        ip = (i+1)*10+(j+1)
        R.matrix[ip,ip-10] = 10.0
        R.matrix[ip,ip-1] = 10.0
        R.matrix[ip,ip+1] = 10.0
        R.matrix[ip,ip+10] = 10.0
        R.matrix[ip,ip-11] = math.sqrt(10.0**2+10.0**2)
        R.matrix[ip,ip-9] = math.sqrt(10.0**2+10.0**2)
        R.matrix[ip,ip+9] = math.sqrt(10.0**2+10.0**2)
        R.matrix[ip,ip+11] = math.sqrt(10.0**2+10.0**2)
```

Example Lecture Assignment #16



Example Lecture Assignment #16

```
for i in range(8):
    for j in range(8):
        ip = (i+1)*10+(j+1)
        if(DEM[i+1,j+1]<=DEM[i,j+1]): R.matrix[ip,ip-10] = 10.0
        else: R.matrix[ip,ip-10] = 10000.0
        if(DEM[i+1,j+1]<=DEM[i+1,j]): R.matrix[ip,ip-1] = 10.0
        else: R.matrix[ip,ip-10] = 10000.0
        if(DEM[i+1,j+1]<=DEM[i+1,j+2]): R.matrix[ip,ip+1] = 10.0
        else: R.matrix[ip,ip-10] = 10000.0
        if(DEM[i+1,j+1]<=DEM[i+2,j]): R.matrix[ip,ip+10] = 10.0
        else: R.matrix[ip,ip-10] = 10000.0
        if(DEM[i+1,j+1]<=DEM[i,j]): R.matrix[ip,ip-11] = math.sqrt(10.0**2+10.0**2)
        else: R.matrix[ip,ip-10] = 10000.0
        if(DEM[i+1,j+1]<=DEM[i,j+2]): R.matrix[ip,ip-9] = math.sqrt(10.0**2+10.0**2)
        else: R.matrix[ip,ip-10] = 10000.0
        if(DEM[i+1,j+1]<=DEM[i+2,j]): R.matrix[ip,ip+9] = math.sqrt(10.0**2+10.0**2)
        else: R.matrix[ip,ip-10] = 10000.0
        if(DEM[i+1,j+1]<=DEM[i+2,j+2]): R.matrix[ip,ip+11] = math.sqrt(10.0**2+10.0**2)
        else: R.matrix[ip,ip-10] = 10000.0
```