Working with Geometries

- Working with geometry objects:
  - Each feature in a feature class contains a set of points that define the vertices of the feature.
  - Geometry token can be used to access the geometry object
  - SHAPE@ will return the full geometry object while SHAPE@XY, SHAPE@LENGTH, etc will return only specific properties of the geometry object.

```
import arcpy
fc:=“c:/data/roads.shp”
cursor=arcpy.da.SearchCursor(fc,[“SHAPE@LENGTH”])
length-0
for row in cursor:
    length+=row[0]
print length
```

- Reading geometries
    - Point feature: each point consists of only a single vertex

```
import arcpy
fc=“c:/data/hospitals.shp”
cursor=arcpy.da.SearchCursor(fc, [“SHAPE@XY”]
for row in cursor:
    x,y=row[0]
    print (“{0},{1}”.format(x,y))
```

- Reading geometries
  - Polyline and polygon features: each feature consists of multiple points, an array of point objects is returned for each feature. To work with these arrays, an extra iteration is needed.

```
import arcpy
from arcpy import env
env.workspace=“c:/data”
fc=“roads.shp”
cursor=arcpy.da.SearchCursor(fc,[“OID@”, “SHAPE@”]
for row in cursor:
   print (“Feature {0}:”.format(row(0))
   for point in row[1].getPart(0):
      print(“{0}, {1}”.format(point.X, point.Y))
```

- Working with multipart features
  - For multipart features, an array containing multiple arrays of point objects is returned, instead of single array of point objects for single-part features.

```
import arcpy
from arcpy import env
env.workspace=“c:/data”
fc=“roads.shp”
cursor=arcpy.SearchCursor(fc,[“OID@”,”SHAPE@”]
for row in cursor:
  print (“Features {0}:”.format(row[0]))
  partnum=0
  for part in row[1]:
    print(“Part{0}:”.format(partnum))
    for point in part:
        print(“{0},{1}”.format(point.X, point.Y))
    partnum+=1
```

- Working with polygons with holes
  - If a polygon contains hole(s), it will consist of a number of rings: one exterior ring and one or more interior rings; A null point object is used as the separator between exterior and interior rings.

```
import arcpy
from arcpy import env
env.workspace=“c:/data”
fc=“roads.shp”
cursor=arcpy.SearchCursor(fc,[“OID@”,”SHAPE@”]
for row in cursor:
  print (“Features {0}:”.format(row[0]))
  partnum=0
  for part in row[1]:
    print(“Part{0}:”.format(partnum))
    for point in part:
        if point:
           print(“{0},{1}”.format(point.X, point.Y))
        else:
           print “Interior Ring”
    partnum+=1
```

- Writing geometries

```
import arcpy, fileinput,os
from arcpy import env
env.workspace=“c:/data”
infile=“c:/data/points.txt”
fc=“newpoly.shp”
Arcpy.CreateFeatureclass_management(“c:/data”,fc, “Polygon”)
cursor=arcpy.da.InsertCursor(fc,[“SHAPE@”])
line_array=arcpy.Array()
point=arcpy.Point()
for line in fileinput.input(infile):
   point.ID, point.X,point.Y=line.split()
   line_array.add(point)
polygon=arcpy.Polygon(array)
cursor.insertRow([polygon])
fileinput.close
Del cur
```