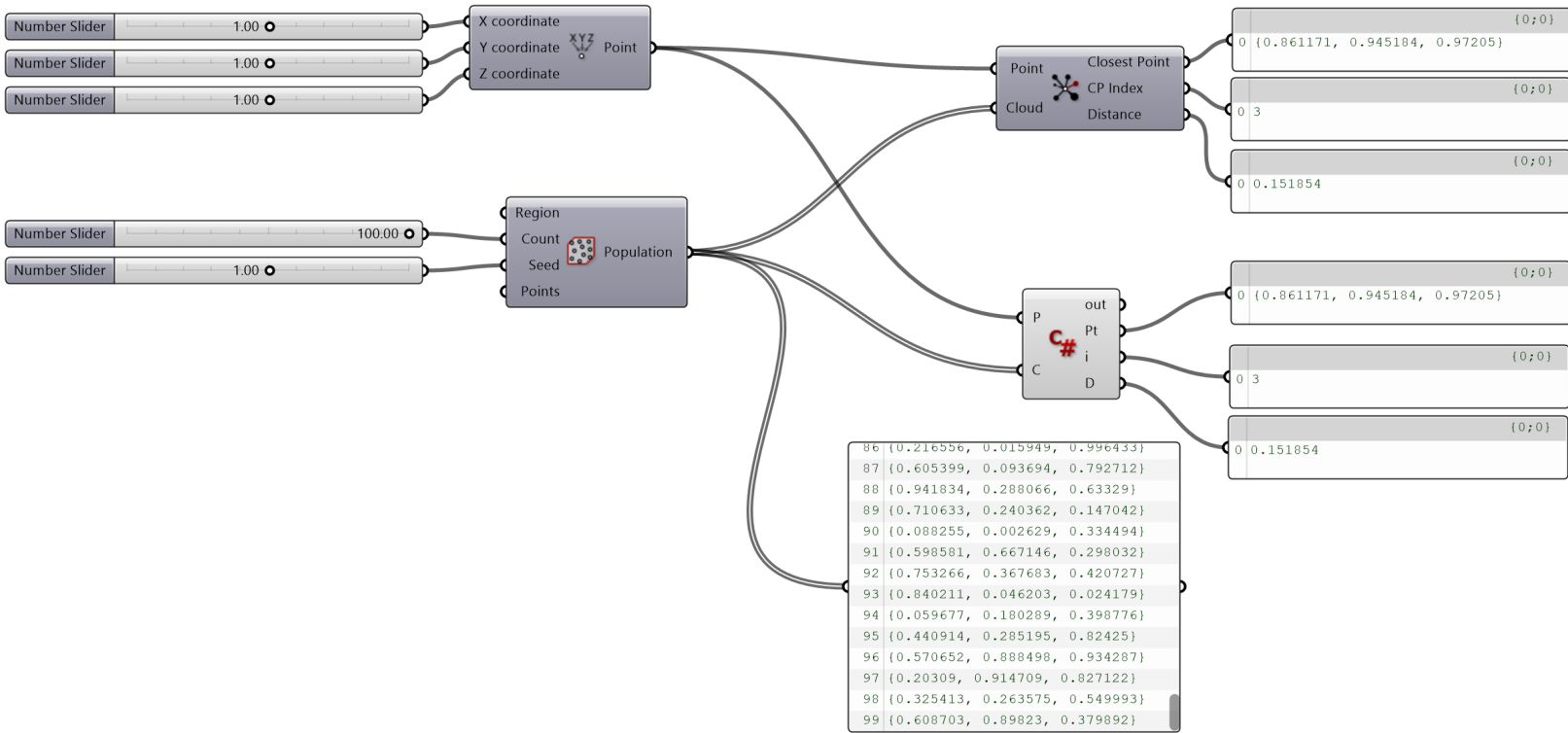


WORKOUT [6]

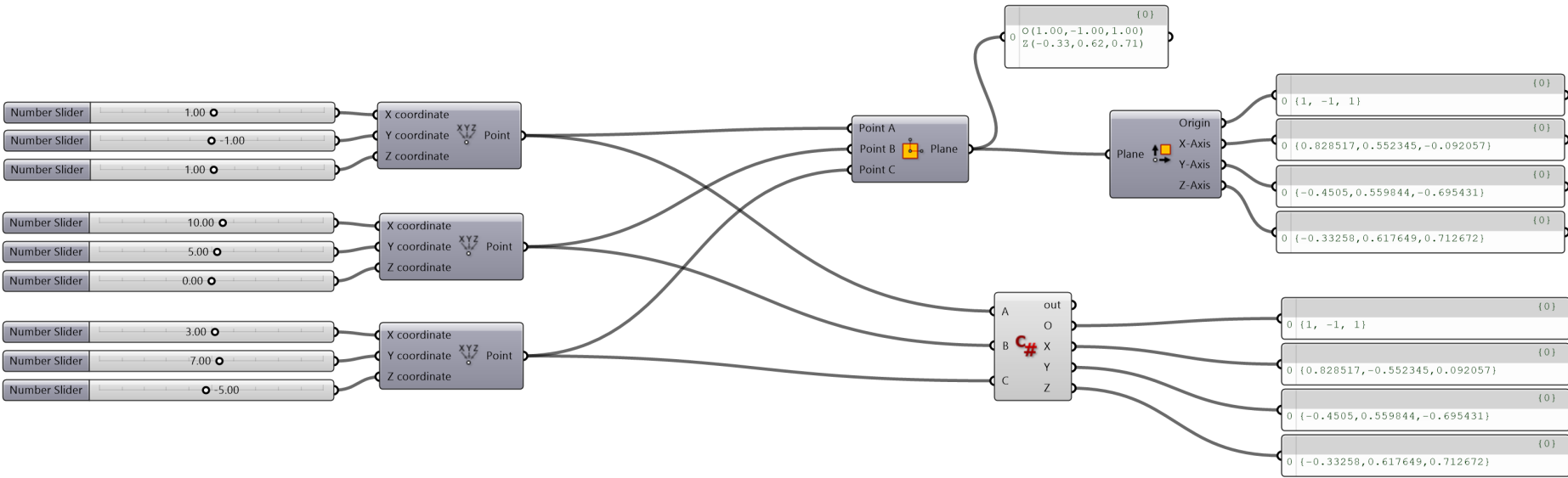
09.28.2020

Find the closest point in a given list to a specific point



```
private void RunScript(Point3d P, List<Point3d> C, ref object Pt, ref object i, ref object D)
{
    double maxD = 100000000000.0;
    int c = C.Count;
    int index = 0;
    for(int a = 0; a < c; a++)
    {
        double d = Math.Sqrt((P.X - C[a].X) * (P.X - C[a].X) + (P.Y - C[a].Y) * (P.Y - C[a].Y) + (P.Z - C[a].Z) * (P.Z - C[a].Z));
        if (d < maxD)
        {
            maxD = d;
            Pt = C[a];
            D = maxD;
            index = index + 1;
            i = index;
        }
    }
}
```

Create a plane with the given three points, then deconstruct the plane without using the plane class



```
private void RunScript(Point3d A, Vector3d B, Vector3d C, ref object O, ref object X, ref object Y, ref object Z)
{
    //z-axis (normal)
    Vector3d AB = new Vector3d(B.X - A.X, B.Y - A.Y, B.Z - A.Z);
    Vector3d AC = new Vector3d(C.X - A.X, C.Y - A.Y, C.Z - A.Z);
    Vector3d z = new Vector3d(AB.Y * AC.Z - AB.Z * AC.Y, -(AB.X * AC.Z - AB.Z * AC.X), AB.X * AC.Y - AB.Y * AC.X);
    double zLength;
    zLength = Math.Sqrt(z.X * z.X + z.Y * z.Y + z.Z * z.Z);
    Vector3d zUnit = new Vector3d(z.X / zLength, z.Y / zLength, z.Z / zLength);

    //Y-axis
    Vector3d y = new Vector3d(z.Y * AB.Z - z.Z * AB.Y, -(z.X * AB.Z - z.Z * AB.X), z.X * AB.Y - z.Y * AB.X);
    double yLength;
    yLength = Math.Sqrt(y.X * y.X + y.Y * y.Y + y.Z * y.Z);
    Vector3d yUnit = new Vector3d(y.X / yLength, y.Y / yLength, y.Z / yLength);

    //X-axis
    Vector3d x = new Vector3d(y.Y * z.Z - y.Z * z.Y, -(z.X * y.Z - z.Z * y.X), z.X * y.Y - z.Y * y.X);
    double xLength;
    xLength = Math.Sqrt(x.X * x.X + x.Y * x.Y + x.Z * x.Z);
    Vector3d xUnit = new Vector3d(x.X / xLength, x.Y / xLength, x.Z / xLength);

    //output
    O = A;
    Z = zUnit;
    Y = yUnit;
    X = xUnit;
}
```