In [ ]:
```
import arcpy
import os
```

Reclassify Slope

In [ ]:
```
out_raster = arcpy.sa.Reclassify(
    in_raster="SlopeLayer",
    reclass_field="VALUE",
    remap="0 2.940712 1;2.940712 6.763637 2;6.763637 11.762846 3;11.762846 21.173123 4;21.173123 74.988144 5",
    missing_values="DATA"
)
out_raster.save(r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Lab2-2.gdb\Reclass_Slope")
```

Reclassify Land Use

In [ ]:
```
out_raster = arcpy.sa.Reclassify(
    in_raster="landusebufferraster",
    reclass_field="NLCD_Land",
    remap="'Open Water' 5;'Developed, Open Space' 1;'Developed, Low Intensity' 1;'Developed, Medium Intensity' 1;'Developed, High Intensity' 1;'Barren Land' 1;'Deciduous Forest' 1;'Evergreen Forest' 1;'Mix
    missing_values="DATA"
)
out_raster.save(r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Lab2-2.gdb\Reclass_LandUse")
```

Combine Rasters with Different Weights

In [ ]:
```
arcpy.env.workspace = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Lab2-2.gdb"
output_folder = r"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\DorysPath.gdb"

# List of input raster names
input_raster_names = ["Reclass_Slope", "Reclass_LandUse"]

#List of Weights
weight_scenarios = [0.25, 0.5]

# Nested loop to process each combination of input rasters and weight scenarios
for raster1_name in input_raster_names:
    for raster2_name in input_raster_names:
        for weight in weight_scenarios:
            if weight == 0.5:
                output_name = f"LandUse_Slope_EqualWeight"
            else:
                output_name = f"{raster1_name}_w{int(weight * 100)}_{raster2_name}_w{int((1 - weight) * 100)}"
            #Skip if loop wants to pair the same rasters together
            if raster1_name == raster2_name:
                continue

            # Paths to rasters
            raster1 = os.path.join(arcpy.env.workspace, raster1_name)
            raster2 = os.path.join(arcpy.env.workspace, raster2_name)

            # Create raster combinations
            raster1_weighted = arcpy.Raster(raster1) * weight
            raster2_weighted = arcpy.Raster(raster2) * (1 - weight)
            output_raster = raster1_weighted + raster2_weighted

            # Save the output raster
            output_raster.save(os.path.join(output_folder, output_name))
            print(f"{output_name} Created and Saved")
```

Calculate Least Cost Path

In [ ]:
```
costsurfaces = ["Reclass_Slope_w25_Reclass_LandUse_w75", "LandUse_Slope_EqualWeight", "Reclass_LandUse_w25_Reclass_Slope_w75"]

for costsurface in costsurfaces:

    arcpy.intelligence.LeastCostPath(
        in_cost_surface=costsurface,
        in_start_point="Least Cost Path Input Starting Point (Points)",
        in_end_point="Least Cost Path Input Ending Point (Points)",
        out_path_feature_class=rf"\\Mac\Home\Documents\ArcGIS\Projects\Lab2-2\Lab2-2.gdb\{costsurface}_LeastCost",
        handle_zeros="NO_DATA"
    )
```