**CACIE Tool #NN** – ***Kingdom to ArcGIS Grid Tool***

kingdom2arcgrid.py

**Version** **1.0**

**QA**: **QA**

1. **Description and Purpose**

The kingdom2arcgrid.py script is a python program that converts Kingdom point files (x, y, z) of surfaces (topographic surface or geologic structure tops) to ASCII raster files (asc) that Leapfrog and ArcGIS can load. Points from Kingdom are not constrained to a rectangular extent, whereas the ASCII raster files are defined for a rectangular domain.

The script scans the Kingdom file first to determine a maximum rectangular domain.

The script then determines the number of rows and columns needed to cover the entire set of Kingdom points (assuming a constant cell size). It then builds the ASCII raster file header which includes:

* Number of columns
* Number of rows
* X-coordinate of the center of the cell at the lower left corner of the rectangular domain
* Y-coordinate of the center of the cell at the lower left corner of the rectangular domain
* Cell size (dx=dy)
* No data value (-99999) for XY locations that do not correspond to a point in the original Kingdom file

Following the header lines, kingdom2arcgrid.py writes out the cell values (surface elevations).

It is noted that the conversion of a Kingdom point file to an ASCII raster file using the kingdom2arcgrid.py script results in a small mismatch between the Kingdom file dy and the ASCII raster file cell size. For the Kingdom point files, dx is constant and dy is constant, although dx and dy are not typically identical. For the ASCII raster files, dx and dy are identical (i.e., cell size describes a square cell). Since kingdom2arcgrid.py uses the distance between the first two X-locations in the Kingdom file to determine dx, the dx for the ASCII raster file is the same as the Kingdom file dx, but the dy values are different. This results in a small location error in the Y-direction that increases with increasing Y values.

1. **Functional Requirements**

The following are the functional requirements of the Kingdom to ArcGIS Grid Tool:

FR-1: Read the Kingdom file data points (X, Y, Z)

FR-2: Determine the X and Y extents required to cover all the Kingdom points; convert to a rectangular domain defined by lower left location (cell center) and the number of columns and number of rows

FR-3: Write the ASCII raster file header lines

FR-4: Write out ASCII raster file cell values one row per line with number of columns per line; row 1 of the data is at the top of the raster, row 2 is just under row 1, and so on; cell values are delimited by spaces

1. **Software Requirements Specifications**

python – version 3.5.2 on a Windows machine (Last modified 1/28/2020)

1. **Software Design Description**

Arguments:

KingdomFile – Name of the Kingdom point file converted to ASCII raster file format.

ASCIIRasterFile – Name of the ASCII raster file output from kingdom2arcgrid.py.

Input Files:

Kingdom point file (comma delimited X, Y, Z point format)

Output file:

ASCII raster file (asc format)

Execution:

The following is the batch script configuration that will be passed as an argument to the Tool Runner for qualified runs:

python kingdom2arcgrid.py {KingdomFile} {ASCIIRasterFile}

Code Review:

Dennis Fryar reviewed the code and associated findings/comments/resolutions are discussed in Appendix B (Table B-1).

1. **Requirements Traceability Matrix**

The requirements traceability matrix for the Kingdom to ArcGIS Grid Tool is presented in Table 1.

| Table 1. Kingdom to ArcGIS Grid Tool Requirements Traceability Matrix | | |
| --- | --- | --- |
| **Functional Requirement ID** | **Acceptance Test ID** | **Test Case** |
| QA Level | CACIE-kingdom2arcgrid.py -IT-1 | Installation Test |
| FR-1  FR-2  FR-3  FR-4 | CACIE- kingdom2arcgrid.py -TC-1 | 1) Inside of bash terminal, execute *kingdom2arcgrid.py\_TC-1.sh* and verify that a new raster file is created in the 200E folder (denoted by a file with the “\*.asc” extension).  2) Verify the asc file header values (suggest using Excel):  a. Number of columns agrees with min/max X range of the Kingdom points.  b. Number of rows agrees with min/max Y range of the Kingdom points.  c. Lower left coordinates are center values and match the minimum X and Y values for the points in the Kingdom file.  d. Cell size is equal to the X point spacing in the Kingdom file.  3) Use ArcGIS Raster to XYZ (or similar tool) to convert the raster files to XYZ format.  4) Use Excel to compare the converted XYZ files back to the original Kingdom .dat files.  5) Evaluate any differences noted in the comparison. |

1. **Test Plan and Cases**

The test plan for the Kingdom to ArcGIS Grid Tool is presented in Table 2.

| Table 2. Kingdom to ArcGIS Grid Tool Installation and Acceptance Test Plan | | | |
| --- | --- | --- | --- |
| **Kingdom to ArcGIS Grid Tool [Installation or Acceptance] Testing**  **CACIE-kingdom2arcgrid.py–[IT or AT]-##** | | **Date:** | |
| **Tool Runner File Location for this test:**  */opt/tools/pylib/runner/runner.py* | | **Test Performed By:** | |
| **Testing Directory:** [**\\olive\backups\CAVE\CA-CIE-Tools-TestEnv\kingdom2arcgrid.py**](file:///\\olive\backups\CAVE\CA-CIE-Tools-TestEnv\kingdom2arcgrid.py) | | | |
| **Test Step** | **Test Instruction** | **Expected Result** | **Test Result  (Pass/Fail)** |
| Tools Code Repository Directory: | | | |
| **CACIE-kingdom2arcgrid.py-IT-1** | | | |
| Navigate to the testing directory | | | |
| 1 | Invoke Tool runner and test the tool using *runner\_run\_IT-1\_kingdom2arcgrid.py.sh* as follows:  *./runner\_run\_IT-1\_kingdom2arcgrid.py.sh* (within a bash terminal) | | |
| 2 | Verify Tool Runner is invoked and executed. | Should see two new files:   1. *runner\_run\_IT-1\_kingdom2arcgrid.py\_logfile.txt* 2. *runner\_run\_IT-1\_kingdom2arcgrid.py\_screen.log* |  |
| **CACIE-kingdom2arcgrid.py–TC-1** | | | |
| Navigate to the Testing Directory | | | |
| 1 | Open a bash terminal and run the test case shell script using the following command: ./kingom2arcgrid.py\_TC-1.sh | A new ASCII raster file should be created in the *./200W* directory. The Tool Runner invocation should resolve to “QUALIFIED” in the *kingdom2arcgrid.py\_TC-1\_logfile.txt* and *kingdom2arcgrid.py* should resolve to “TEST” |  |
| 2 | Verify the asc file header values (suggest using Excel):  a. Number of columns agrees with min/max X range of the Kingdom points.  b. Number of rows agrees with min/max Y range of the Kingdom points.  c. Lower left coordinates are center values and match the minimum X and Y values for the points in the Kingdom file.  d. Cell size is equal to the X point spacing in the Kingdom file. | Agreement between Kingdom point file dimensions and ASCII raster file headers. The Kingdom point file to compare with is in *./200W/CPVZ\_200West\_Rev1\_Topo.asc* |  |
| 3 | Use ArcGIS Raster to XYZ (or similar tool) to convert the raster file to XYZ format. Use the following (for ArcMap)   * Enable “Spatial Analyst” extension (if already enabled, continue to the next step) * In the Catalog, open “Toolboxes/System Toolboxes/Spatial Analyst Tools.tbx/Extraction/Sample” * Assign the *CPVZ\_200West\_Rev1\_Topo.asc* to the “Input rasters” and “Input location raster or point features” fields of the tool * Save the file *under. /200W/CPVZ\_200West\_Rev1\_Topo\_pnts* * In ArcMap, under the “Table of Contents”, right-click on the newly created layer/table and click “Open” * In the “Table Options” dropdown ( ) select “Export” * Select Export to “All records” * Click the  and set “Save As Type” to “Text File” and save it as a new file under *./200W/CPVZ\_200East\_Rev1\_Topo\_pnts* | | |
| 4 | Use Excel to compare the converted XYZ files back to the original Kingdom .dat files. | | |
| 5 | Evaluate any differences noted in the comparison. | No significant differences in X, Y or Z except for the known limitation of the tool with respect to the values in the Y-direction as described in Section 1 of this attachment. Collect the absolute value of the maximum difference for the X, Y, and Z coordinates. Differences should be no greater than the following:   |  |  |  | | --- | --- | --- | | **X** | **Y** | **Z** | | 6E04 | 4.0 | 8E-06 | |  |

1. **Acceptance Test Report**

The results of the acceptance test case are as follows:

* + Acceptance Test 1 is in Table A-1. The test passed without errors.
* The test case was performed by Jacob Fullerton on February 4th, 2020. For details, please refer to Appendix A.

1. **User Guide**

To run this tool, Python need to be installed. Once its installed, the tool can be executed with following path in a command window

python kingdom2arcgrid.py rid.py {Kingdom point (input) file} {ASCII Grid (output) file}

Need access to the input files described in Section 4: Software Design description under input files.

For additional information how to run this tool check Section 4 (Software Design Description)

**Appendix A**

**Acceptance Testing Logs**

| **Table A-1.**  **kingdom2arcgrid Acceptance Test Case 1** | | | |
| --- | --- | --- | --- |
| **Kingdom to ArcGIS Grid Tool Acceptance Testing**  **CACIE-kingdom2arcgrid-AT-1** | | **Date: February 4th, 2020** | |
| **Tool Runner File Location for this test:**  [*\\olive\backups\CAVE\CA-CIE-Tools-TestEnv\kingdom2arcgrid.py*](file:///\\olive\backups\CAVE\CA-CIE-Tools-TestEnv\kingdom2arcgrid.py)**Filename:**  kingdom2arcgrid.py\_TC-1\_logfile.txt | | **Test Performed By: Jacob Fullerton** | |
| **Testing Directory:** [**\\olive\backups\CAVE\CA-CIE-Tools-TestEnv\kingdom2arcgrid.py**](file:///\\olive\backups\CAVE\CA-CIE-Tools-TestEnv\kingdom2arcgrid.py) | | | |
| **Test Step** | **Test Instruction** | **Expected Result** | **Test Result  (Pass/Fail)** |
| Navigate to the Testing Directory | | | |
| 1 | Open a bash terminal and run the test case shell script using the following command: *./kingom2arcgrid.py\_TC-1.sh* | A new ASCII raster file should be created in the *./200W* directory. The Tool Runner invocation should resolve to “QUALIFIED” in the *kingdom2arcgrid.py\_TC-1\_logfile.txt* and *kingdom2arcgrid.py* should resolve to “TEST” | PASS; See Figure 1 |
| 2 | Verify the asc file header values (suggest using Excel):  a. Number of columns agrees with min/max X range of the Kingdom points.  b. Number of rows agrees with min/max Y range of the Kingdom points.  c. Lower left coordinates are center values and match the minimum X and Y values for the points in the Kingdom file.  d. Cell size is equal to the X point spacing in the Kingdom file. | Agreement between Kingdom point file dimensions and ASCII raster file headers. The Kingdom point file to compare with is in *./200W/CPVZ\_200West\_Rev1\_Topo.asc* | PASS; See Table 3 |
| 3 | Use ArcGIS Raster to XYZ (or similar tool) to convert the raster file to XYZ format. Use the following (for ArcMap)   * Enable “Spatial Analyst” extension (if already enabled, continue to next step) * In the Catalog, open “Toolboxes/System Tooboxes/Spatial Analyst Tools.tbx/Extraction/Sample” * Assign the *CPVZ\_200West\_Rev1\_Topo.asc* to the “Input rasters” and “Input location raster or point features” fields of the tool * Save the file under *./200W/CPVZ\_200West\_Rev1\_Topo\_pnts* * In ArcMap, under the “Table of Contents”, right-click on the newly created layer/table and click “Open” * In the “Table Options” dropdown ( ) select “Export” * Select Export to “All records”   Click the  and set “Save As Type” to “Text File” and save it as a new file under *./200W* | | |
| 4 | Use Excel to compare the converted XYZ files back to the original Kingdom dat files. | | |
| 5 | Evaluate any differences noted in the comparison. | No significant differences in X, Y or Z except for the known limitation of the tool with respect to the values in the Y-direction as described in Section 1 of this attachment. Collect the absolute value of the maximum difference for the X, Y, and Z coordinates. Differences should be no greater than the following:   |  |  |  | | --- | --- | --- | | **X** | **Y** | **Z** | | 6E-04 | 4.0 | 8E-06 | | PASS; see Table 4 |

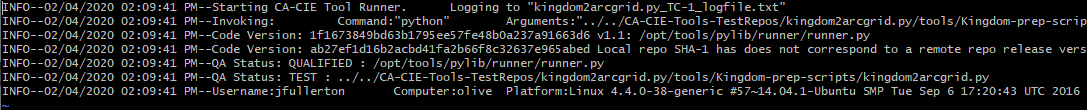


Figure 1. Tool Runner Log

Table 3. TC-1 Step 2 Acceptance Test Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Raster** | |  | **Kingdom Points File** | |
| ncols | 854 |  | Number of Points | 766711 |
| nrows | 901 |  | Delta-X | 10.010428 |
| X\_Min (on-cell-center) | 563579.4337 |  | X\_Min | 563579.4337 |
| Y\_Min (on-cell-center) | 132685.8724 |  | Y\_Min | 132685.8724 |
| X\_Max (on-cell-center)\* | 572118.328778 |  | X\_Max | 572118.3293 |
| Y\_Max (on-cell-center)\* | 141695.257625 |  | Y\_Max | 141691.6237 |
| cellsize | 10.010428 |  |  |  |
| Number\_of\_Cells | 769454 |  |  |  |
| |  | | --- | |  | |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| **Acceptance Test (Step 2)** | | | | |
| Parts (a), (b) and (c) | "Raster" (X\_Min, Y\_Min) are equal to the "Kingdom Points" (X\_Min, Y\_Min). "Raster" (X\_Max, Y\_Max) is greater than "Kingdom Points" (X\_Max, Y\_Max). | | | |
|
|
| Part (b) | Delta-X of "Kingdom Points" is equal to cellsize of "Raster" | | | |

Table 4. TC-1 Step 5 Acceptance Test Results

|  |  |  |
| --- | --- | --- |
| **Max Delta X** | **Max Delta Y** | **Max Delta Z** |
| 5.13E-04 | 3.633972 | 7.6172E-06 |

**Appendix B**

**Code Review Summary**

|  |  |  |
| --- | --- | --- |
| **Table B-1. kingdom2arcgrid.py**  **Code Review Summary** | | |
| **Code Line** | **Comment** | **Resolution** |
| 93 | The conversion of a Kingdom point file to an ASCII raster file using the kingdom2arcgrid.py script results in a small mismatch between the Kingdom file dy and the ASCII raster file cell size. For the Kingdom point files, dx is constant and dy is constant, although dx and dy are not typically identical. For the ASCII raster files, dx and dy are identical (i.e., cell size describes a square cell). Since kingdom2arcgrid.py uses the distance between the first two X-locations in the Kingdom file to determine dx, the dx for the ASCII raster file is the same as the Kingdom file dx, but the dy values are different. This results in a small location error in the Y-direction that increases with increasing Y values. | The Y location differences were not considered large enough to adversely impact the intended used of the ASCII raster files. |

**Appendix C**

Installation Test Log

**Table C-1.**

**kingdom2arcgrid Installation Test**

|  |  |  |  |
| --- | --- | --- | --- |
| **Kingdom to ArcGIS Grid Tool [Installation or Acceptance] Testing**  **CACIE-kingdom2arcgrid.py–[IT or AT]-##** | | **Date:** | |
| **Tool Runner File Location for this test:** | | **Test Performed By:** | |
| **Testing Directory: \\olive\backups\CAVE\CA-CIE-Tools-TestEnv\kingdom2arcgrid.py** | | | |
| **Test Step** | **Test Instruction** | **Expected Result** | **Test Result**  **(Pass/Fail)** |
| Tools Code Repository Directory: | | | |
| Navigate to the testing directory | | | |
| 1 | Invoke Tool runner and test the tool using *runner\_run\_IT-1\_kingdom2arcgrid.py.sh* as follows:  *./runner\_run\_IT-1\_kingdom2arcgrid.py.sh* (within a bash terminal) | | |
| 2 | Verify Tool Runner is invoked and executed. | Should see two new files:   1. *runner\_run\_IT-1\_kingdom2arcgrid.py\_logfile.txt* 2. *runner\_run\_IT-1\_kingdom2arcgrid.py\_screen.log* |  |