CSE360 SEMESTER PROJECT

PHASE 2 SUBMISSION: NETWORK PATH ANALYZER USER GUIDE

Team #9
Team Members:
Dressman, Brandon
Sneberger, Michael
VanDiepenbos, Scott

Submitted via Blackboard:
by November 5, 2018

Contents

Introduction:

The NETWORK PATH ANALYZER is a program designed to allow user to enter various project activities along with their durations and dependencies (activities which must occur prior) into the program which will then calculate the various work paths that could possibly flow through the project – particularly identifying the critical or shortest path.

This User Guide will explain the topics outlined in the Table-of-Contents including: system requirements, installation, how to start the program and input data including any limitations on the forms of the entered data and what an error might look like if the data is entered incorrectly, how to run the program either without making a report of the results or making a report of the results including how to alter already-entered data and rerun the program and how to delete entered data and end the program. Examples runs of the program are provided and various error messages you might encounter are identified and explained.

Overview:

To use the program a user inputs their data (activities/durations/dependencies) in a presented table. An unlimited number of activities is allowed. If any data is entered incorrectly an error message is shown and re-entry requested. After running the program, output is either presented in table format showing all various activity paths through the project, or is provided to you by way of a report in the form of a .txt file with the critical path being the first pathway shown. If the project as entered allows for a "cycle" or looping back upon itself, an error message will be shown as cycles are not permitted.

System Requirements and Installation:

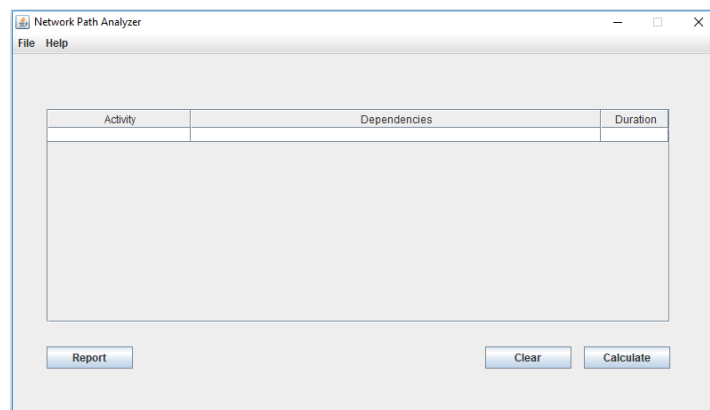## Requirements for running the program:

An active JAVA RUNTIME ENVIRONMENT (present in most computers) and the program's .exe file.

Installation:

No formal installation is required. User merely double-clicks the .exe file to initiate the program.

Getting Started:

Upon opening the .exe file user will be presented with an Entry Table that appears as follows:

Input:

Once presented with the Entry Table, user will enter the various activities comprising the project by entering the following areas of information into the appropriate cell of the Entry Table:

- activity name as a string of characters which can include numbers or be totally numerical;
- duration of the activity as an integer (example: 1, 5, 13, 22) (*see* example error below);
- list of dependencies (predecessors), in other words what activity or activities must be completed prior to activity. Note that a listed dependency must be entered exactly as the activity was initially entered or an error will be shown (*see* example error below).
- Each of these data entries can be changed before or after the program is run.

Data Entry Limitations:

Note that all activities must be connected to another activity, in other words each activity must either serve as a predecessor or have a predecessor. The starting activity (there may be more than one starting activity) must not depend on any prior activity, and the final activity must not serve as a predecessor to any activity.

Maximum Number of Activities:

There is no maximum number of activities and/or dependencies(predecessors).

Error Detection:

If an error is detected in user's input, user will be notified and re-prompted for new input (*see* example error below).

Running the Program with or without a Written Report, Ending the Program:

Running the Program:

Once user has completed their input, there are two different ways to run the program:

- click on the [Calculate] button on the lower right of the Entry Table frame to begin processing and generate an on-screen table of results; or
- click on the [Report] button on the lower left of the Entry Table frame after which the program will perform the calculation and provide you with a written report of the results in the form of a text file.

Re-Running the Program without Exiting Completely:

After the program has run user has two options for re-running the Program:

- If the user ran the program by clicking on the [Calculate] button, to rerun they can click the [Close] button on the lower right of the Results Table frame to close the results frame and return to the Entry Table;
- If the user ran the program by Clicking the [Report] button, to rerun then once the report is generated the user will still be presented with the Entry Table;
  - once either of the above is done, the can retain and/or modify the data currently entered in the Entry Table and modify it as needed by double-clicking inside the cell to be edited; or
  - user can delete the currently-entered data by clicking the [Clear] button data and restart the data entry process.
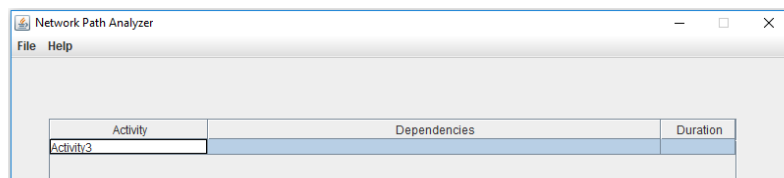
Ending the Program and Exiting Completely:

Once user has completed their input and run the program either a Results Table or Report will be generated. To end the Program:

- if the user ran the program by clicking the [Calculate] button: click the [Close] button on the Results Frame to close the frame; then however the user ran the program
- click the [X] in the upper-right corner of the entry frame to end the program.
- to restart the program after exiting completely, double-click the original .exe file.
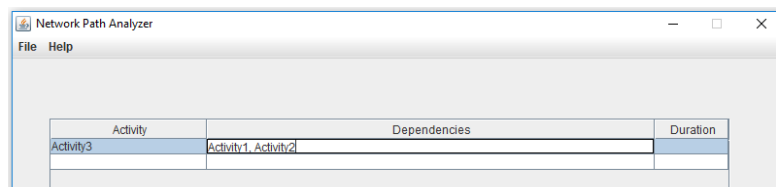
Example Run of the Program Number 1:

Entering Path Information:

Start by entering the name of an activity (you do not need to start with the first activity of the project) into the Entry Table. Activity names can have letters, numbers and characters, but no spaces (example allowable activity names: A, Charlie, D23, Foxtrot-5, Golf_19, 625).

Next, enter any dependencies this activity will have. If more than one dependency, separate entries using commas. If no dependencies are entered for a particular activity, then that activity is assumed to be the start activity. Note that dependencies must match their associated activity names exactly. If they do not, an error message indicating there is not such activity name will be shown (*see* example error below).
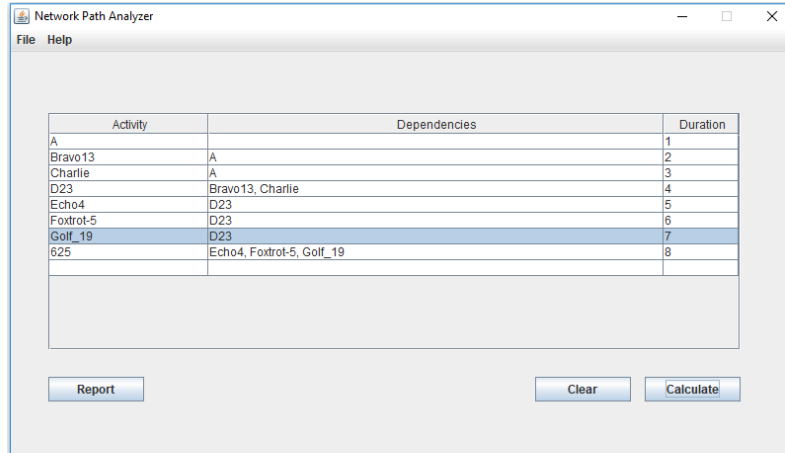
Finally, enter the duration value rounded up to the nearest integer.

Calculating Paths/Output:

After entering all Activities with their corresponding Dependencies and durations you are now ready to calculate all available paths. For an example, let us assume you entered the following data (note the different allowable forms of activity name):

Once data is properly entered, there are two ways to have the Program calculate the various paths through you project:

1) Click the [Calculate] button at the bottom-right of the Entry Table frame

The Program will output the following: a list of all possible activity paths through the project, with the Critical Path being listed first. Paths after the critical path are listed in descending order by duration. The critical path will be the first path displayed on the list. The data in this example is fine, but if you enter a project that is improper you will see an error message (*see* example error below).

If you want the program to show the Critical Path only, in the results table click the [Show Critical Path] check box and all results but the critical path will be masked.

```
Path Analysis                                                          ×

    ┌─────────────────────────────────────────────┬──────────────┐
    │                    Path                     │   Duration   │
    ├─────────────────────────────────────────────┼──────────────┤
    │ A,Charlie,D23,Golf_19,625                   │ 23           │
    │                                             │              │
    │                                             │              │
    │                                             │              │
    └─────────────────────────────────────────────┴──────────────┘

    ☑ Show Critical Path                              [   Close   ]
```

2) Click the [Report] button at the bottom-left of the Entry Table frame

An alternative way to run the Program also allows you to save the data you enter and the result of the program's analysis by creating a report. The report will be a text file that you can name that will be as follows, including all of the data you entered and all calculated paths through the project:

```
NETWORK DIAGRAM "User's Unique Name Here"

Activities                                                    Durations
---------------------------------------------------------------------------
A                                                                 1
Bravo13                                                           2
Charlie                                                           3
D23                                                               4
Echo4                                                             5
Foxtrot-5                                                         6
Golf_19                                                           7
625                                                               8


---------------------------------------------------------------------------

Paths
---------------------------------------------------------------------------
Path:      A,Charlie,D23,Golf_19,625
Duration:  23

Path:      A,Bravo13,D23,Golf_19,625
Duration:  22

Path:      A,Charlie,D23,Foxtrot-5,625
Duration:  22

Path:      A,Bravo13,D23,Foxtrot-5,625
Duration:  21

Path:      A,Charlie,D23,Echo4,625
Duration:  21

Path:      A,Bravo13,D23,Echo4,625
Duration:  20

---------------------------------------------------------------------------
                                              Created On: 01/01/2019
```
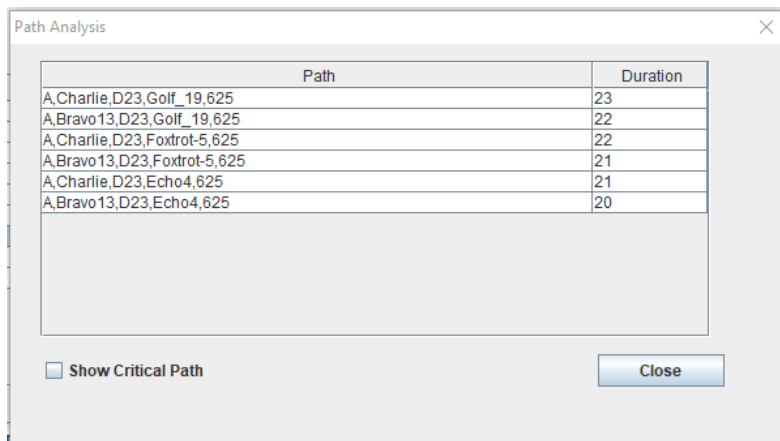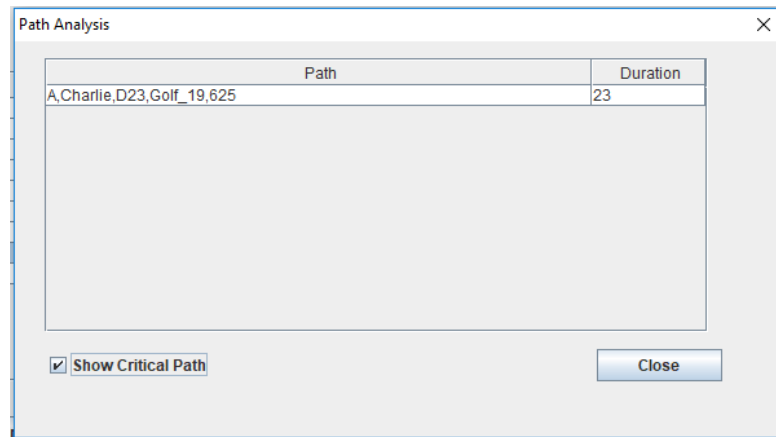
Example Run of the Program Number 2:

Entering Path Information:

Here is a how an alternative entry of data might appear:

Activity	Duration	Dependencies
A	3	
B	4	A
C	2	A
D	5	B
E	1	C
F	2	C
G	4	D,E
H	3	F,G

[Clear]   [Calculate]

Calculating Paths/Output:

After entering all Activities with their corresponding Duration and Dependencies as show above user is now ready to calculate all available paths. Click the [Calculate] button at the bottom of the screen and the Program will output the following: a list of all possible activity paths through the project, with the critical or fastest path being listed first. Paths after the critical path are listed in descending order by duration. Note that the critical path – or path with the longest overall duration is listed first:

Path	Total Duration
A,B,D,G,H	19
A,C,E,G,H	13
A,C,F,H	10

[Close]

Re-Running the Program:

To re-run the program (you might like to do this if you want to adjust the data you have input without starting over), if you ran the program by clicking the [Calculate] button you simply click the [Close] button of the result frame. This will close the results frame but leave the initial data entry frame open. If you ran the program by clicking the [Report] button, once the report is generated you will be back at the Entry Table. Now that you are back to the Entry Table you have multiple options:

- You can alter the data you already entered by double-clicking int the cell you want to change, then deleting the old data and entering new data as needed, or
- You can delete all the data you entered previously by clicking the [Clear] button and starting over by entering new data.

Once you have adjusted your entered data, you can rerun the program either by clicking the [Calculate] button or the [Report] button on the data entry frame. You can continue this loop of adjusting your data and rerunning the program as many times as you like.

# Examples of How Error Messages Appears:

Error if A dependency Name Does Not Match An Activity Name:

Based on Example Run of the Program 1 above, note the following example where the activity name "Fostrot-5" is properly entered, but the reference so that activity as a dependency is mis-entered as "Fostrot-5" the error appears as follows:
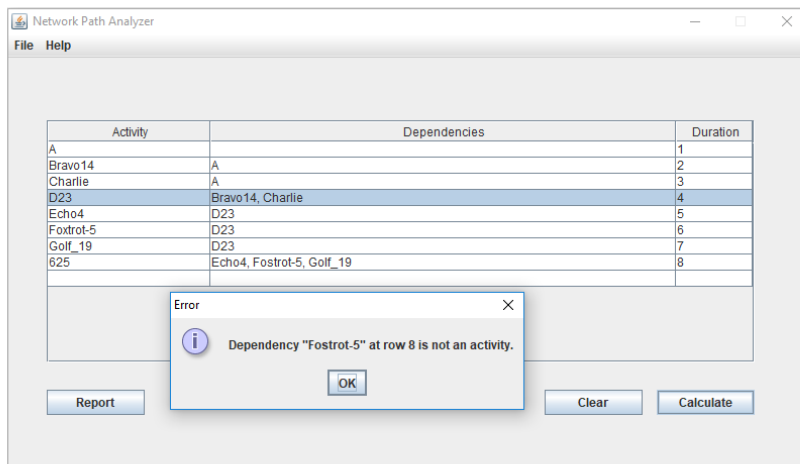

Error if a Duration is Not Entered as an Integer:

If a duration is not entered as an integer – for example as 4.2 here – the following error will display showing that the duration is invalid:
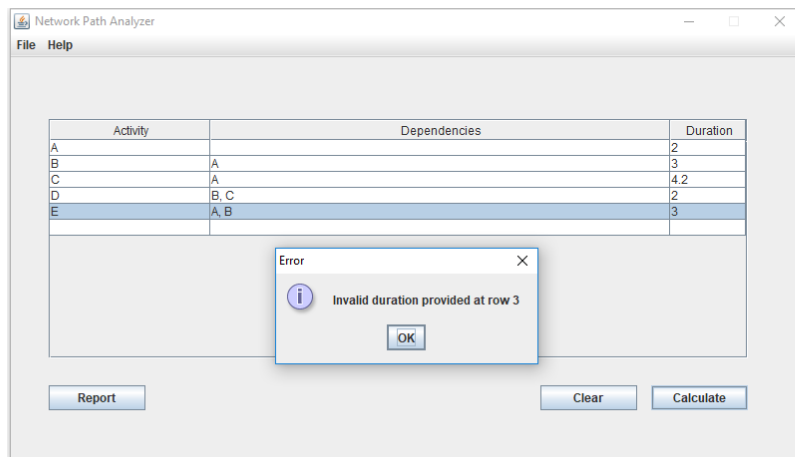
Error if The Project as Entered Features a Cycle:

As instructed, cycles are not allowed in entered projects. If upon analysis a project is found to contain a cycle, an error like the following will be shown:
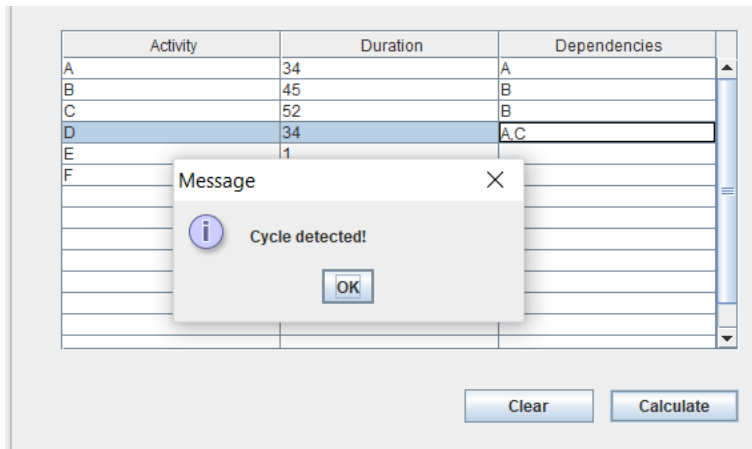
Error if The Project as Entered Is Not Connected:

With exception of the start activity that has no dependency or predecessor, and the end activities that do not serve as a dependency or predecessor to any other activity, each of the activities of a project must both have at least one activity preceding it, and at least one activity following it. In the following data note that activity E has activities A and B as dependencies, but no other activity follows activity E. This means the project as entered is not connected and an errors would be shown when the program was run (note that activity E is not a second end activity because it does not depend on activity C):
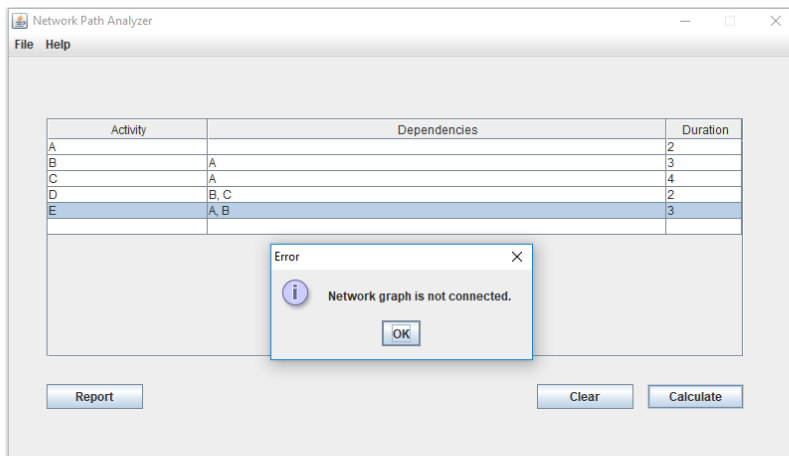
Conclusion:

The creators of the NETWORK PATH ANALYZER are pleased to provide users with an easy, efficient way to identify critical pathways in projects of all sizes. Users enter the various activities and the program performs the mathematical organization and calculation required to find the critical path through the project with the option of printing out a written report.