Assignment 2 – Traffic-based Route Guidance System

- **Due:**
  - **Part A** *11:59pm Sunday 13 April 2025 (Week 6)*
  - **Part B** *11:59pm Sunday 25 May 2025 (Week 11)*
  - **Part C** *11:59pm Sunday 1 June 2025 (Week 12)*

- **Contributes** 50% to your final subject result, subject to moderation if required.
- **Group of 3-4 students**.

*This assignment project requires you to work as part of a group to design and develop a traffic-based route guidance system, **TBRGS**.*

This project requires your group to implement and demonstrate a traffic-based route guidance system, TBRGS, that provides route guidance based on the prediction of traffic. Accurate and timely traffic flow information is important for traffic authorities (such as VicRoads) to identify congested areas and implement traffic management policies to reduce congestion. It is also critical for route guidance systems (such as Google maps) to calculate the best routes for their users considering the traffic conditions potentially experienced along each route. Thanks to recent advancement in sensor technologies, traffic authorities are able to collect massive amount of traffic data to enable accurate predictions of traffic information (such as traffic flow, speed, travel time, etc.). Your group will be required to develop a solution to utilise historical traffic data for traffic flow prediction.

The TBRGS system will use a small dataset from VicRoads for the city of Boroondara that contains only the traffic flow data (the number of cars passing an intersection every 15 minutes) to train a machine learning (ML) model for traffic flow prediction. Your task is to implement machine learning (ML) algorithms that takes an input traffic flow dataset in a predefined format (to be specified by you) and train machine learning models to allow relevant traffic conditions at specified time to be predicted. Furthermore, the TBRGS system can also calculate alternative routes to allow a driver to travel from an origin (O) to their destination (D) and recommend the path that takes the least amount of time.

This assignment project will be structured into three parts:

- **Part A:** You are tasked with implementing tree-based search algorithms in software (from scratch) to search for solutions to the ***Route Finding*** problem.
- **Part B:** You are tasked with implementing machine learning (ML) algorithms that takes an input traffic flow dataset in a predefined format (to be specified by you) and train machine learning models to allow relevant traffic conditions at specified time to be predicted. Yu are also required to integrate Part A and Part B to allow TBRGS to calculate the optimal route between O and D by taking into account the travel time based on the traffic condition predicted by your ML model.
- **Part C:** You are required to investigate the ethical considerations when designing and implementing TBRGS to understand the potential impacts of different design decisions on ethical issues related to TBRGS.

Detailed specifications of each part will follow.

Project requirements:

- Source code maintained on Git based VCS (Github/Bitbucket/GitLab/…). You must provide read-only access to the tutor/lecturer.
- Running illustrative demo of a working prototype (please refer to **Marking Schemes** in each part for details on functionality that needs to be implemented)

Part A – Tree Based Search

- **Due** *11:59pm Sunday 13 April 2025 (Week 6)*
- **Contributes** 40% to Assignment 2 result.

Part A of Assignment 2 (2A) requires you to work as part of your group to get working tree based search algorithms.

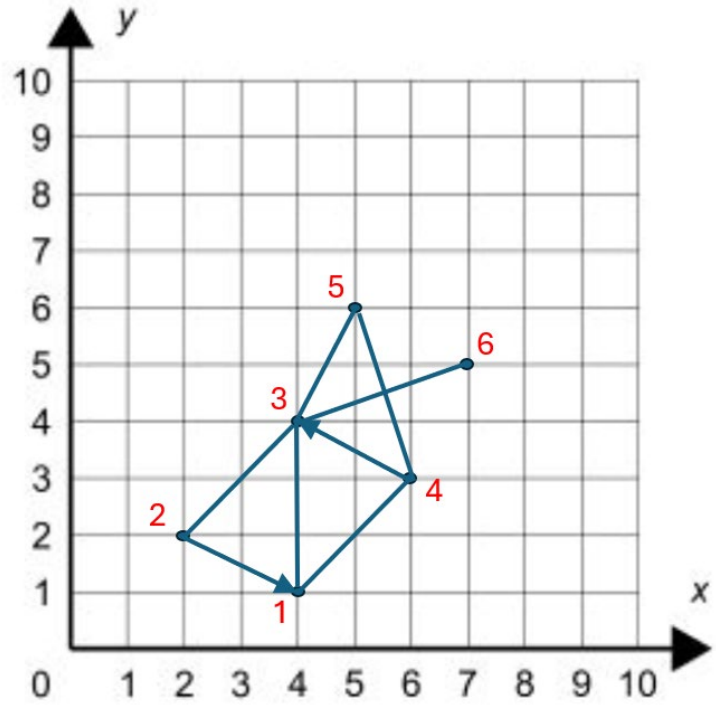
You need to implement tree-based search algorithms in software (from scratch) to search for solutions to the ***Route Finding*** problem. Both ***informed*** and ***uninformed*** methods will be required. You will also need to do some self-learning to learn several search methods (not covered in the lectures).

Implementation

You are encouraged to implement your software using Python. If you prefer to implement your assignment in one of the alternative languages Java, C++ or C#, you need to discuss with your tutor to seek their permission. **You must gain permission from the convenor before using anything else.** Assignment work will be tested on a standard **Microsoft Windows 10** system.

The Route Finding Problem

In this problem, our agent is tasked with finding optimal paths (i.e., lowest cost) between from an Origin node (O) to some Destination (D) nodes on a 2D graph. For example, the following graph represents one such problem:



In the above graph, some edges with an arrow indicate that they can only be traversed in one direction. For instance, one can traverse directly from Node 2 to Node 1 but one **cannot traverse directly** from Node 1 to Node 2. The above *graph* (also known as a *problem*) can be expressed by the following specification:

Nodes:
1: (4,1)
2: (2,2)
3: (4,4)
4: (6,3)
5: (5,6)
6: (7,5)
Edges:

(2,1): 4
(3,1): 5
(1,3): 5
(2,3): 4
(3,2): 5
(4,1): 6
(1,4): 6
(4,3): 5
(3,5): 6
(5,3): 6
(4,5): 7
(5,4): 8
(6,3): 7
(3,6): 7
Origin:
2
Destinations:
5; 4

File Format: The problems are stored in simple text files with the following format:

- A set of nodes following the keyword **"Nodes:"** ; each line following this keyword specifies a node and its coordinates; for example, **1: (4,1)** specifies a node **1** whose coordinate is **(4,1)** with **4** being the x-coordinate and **1** being the y-coordinate of this node;
- A set of edges connecting the nodes following the keyword **"Edges:"**; each line following this keyword specifies an edge and its cost; for example **(2,1): 4** specifies an edge allowing you to traverse from node **2** to node **1** with a cost of **4**;
- The origin node following the keyword **"Origin:"**;
- A set of destination nodes following the keyword **"Destinations:".** The destination nodes are separated by semi-colons (;). There can be 1, 2, or multiple nodes in the set of destination nodes.

Search Algorithms

The following describe a number of tree based search algorithms. DFS, BFS, GBFS and AS have been covered in the lectures and the tutorials. CUS1 and CUS2 are two algorithms you may learn by yourself (from the textbook, from the Internet or any other sources).

NOTE 1: The objective is to reach one of the destination nodes.

NOTE 2: When all else is equal, nodes should be expanded according to the ascending order, i.e., from the smaller to the bigger nodes. For instance, when all else are being equal between nodes 4 and 7, Node 4 should be expanded before Node 7. Furthermore, **when all else is equal**, the two nodes N_1 and N_2 **on two different branches** of the search tree should be expanded according to the chronological order: if node N_1 is added BEFORE node N_2 then N_1 is expanded BEFORE node N_2.

Search Strategy	Description	Method
Uninformed		
depth-first search	Select one option, try it, go back when there are no more options	DFS
breadth-first search	Expand all options one level at a time	BFS
Informed		
greedy best-first	Use only the cost to reach the goal from the current node to evaluate the node	GBFS
A* ("A Star")	Use both the cost to reach the goal from the current node and the cost to reach this node to evaluate the node	AS
Custom		
Your search strategy 1	An **uninformed** method to find a path to reach the goal.	CUS1
Your search strategy 2	An **informed** method to find a shortest path (with least moves) to reach the goal.	CUS2

Command Line Operation

Your program needs to operate from a DOS command-line interface to support batch testing. A DOS command-line interface can be brought up in Windows 7/8/10 by typing **cmd** into the search box at the ***Start*** button. However, please ensure that your program works on Windows 10 because we will be testing your program on Windows 10. This can be accomplished with a simple DOS .bat (batch) file if needed. Below are the three different arguments formats you need to support. Note the unique argument count for each.

```
C:\Assignments> search <filename> <method>
```

where **`search`** is your .exe file or a .bat (batch) file that calls your program with the parameters.

(if you program in Python, we can accept Python scripts and execute your scripts using the following command from the CLI:

```
C:\Assignments> python search.py <filename> <method> )
```

When a goal can be reached, standard output needs to be in the following format:

```
filename method
goal number_of_nodes
path
```

where **`goal`** is the goal node your search method reached and **`number_of_nodes`** is the number of nodes your program has created when perform this search strategy, and **`path`** is a sequence of moves in the solution that brings you from the start-configuration to the end-configuration. Line breaks are ignored (so use them if you want to).

Report file

You must also include a report which has to be either in Microsoft Word or in PDF whose name is your Team ID (for example, 123.PDF) containing your report. The aim of this report is for you to summarise your understanding of the problem and the algorithms you have used to solve the problem. We are most interested in the insights you have gained, especially those **using data obtained by running your software**.

Report Details: The report must be between 8 and 10 pages (excluding cover page and TOC).

- **Cover page:** including details of all group members (i.e., full names and student IDs), signed by all members of the group.
- **Table of contents (TOC)**.
- **Instructions:** Basic instructions of how to use your program. You can also include a **note** containing anything else you want to tell the marker, such as how to use the GUI version of your program, and something particular about your implementation.
- **Introduction:** Introduce the *Route Finding Problem*, basic graph and tree concepts and concise information about the algorithms the team have implemented.
- **Features/Bugs/Missing:** Include a list of the features you have implemented. Clearly state if a required feature has not been implemented. Failure to do this will result in penalties. Include a list of any known bugs. Also, anything else you want to tell the marker, such as how to use the GUI version of your program, and something particular about your implementation.
- **Testing:** Provide an overview of the test cases you have created to test your program (either manually or automatically). Report the results of testing your program.
- **Insights:** Present and discuss the qualities of the search algorithms used in your assignment. Which algorithms are better and why? **Use data collected to support your points.**
- **Research (if applicable):** If you managed to do some additional research to improve the program in some ways, please report it here.
- **Conclusion:** Conclude with a discussion about the best type of search algorithm you would use for this type of problem. Include thoughts about how you could improve performance.
- **Acknowledgements/Resources:** Include in your report a list of the resources you have used to create your work. A simple list of URL's is not enough. Include with each entry a basic description of how the person or website assisted you in your work.
- **References:** Cite the sources you referred to in your Assignment (implementation, report, etc.)

Tips:

- All figures and tables need to be properly captioned with sensible descriptions.
- Report presentation should include header/footer information (pages numbers, etc.)

Marking Scheme:

Requirements (or equivalent sections)	***Mark***
For each of the 6 methods Depth-First Search (DFS), Breadth-First Search (BFS), greedy best-first search (GBFS), A* (AS), CUS1 and CUS2, if you can get it work well	**10(x6)**
Testing: At least 10 test cases have been created to cover different problem scenarios. Test results have been checked and documented.	**10**
Report: Clear and provide sufficient information about the problem and your algorithm/solution AND the insights you have obtained through your program.	**20**
Research: If you show some initiatives in researching about the problem and solutions, or carrying out extensive tests to provide interesting data about the algorithms, or getting some clever optimization, etc. with a well-written **Research** section in the report to discuss and demonstrate these initiatives	**10**
Total	**100**
You need to follow good programming practice (e.g., well-designed, well-structure codes with clear and helpful comments). Failure to do so get penalty.	**Up to -10**

For each individual student: (the maximum mark a student can get for Assignment 2A is 100 and the minimum mark is 0)	***Mark***
Failure to provide in class updates of the assignment progress to your tutor.	**Up to -40**
Failure to contribute to the work performed by the team or to collaborate with other team members to achieve the desired outcomes for the project.	**Up to -80**

An idea for research initiatives

- Can you modify your program so that the robot will visit ALL destination nodes with the SHORTEST path? Please include in your report the challenges you had to overcome to address this requirement and how you solved them.
- For other ideas, please feel free to do the research yourselves and discuss with your tutors to get their approval first before doing the research component.

Part B – Machine Learning and Software Integration

- **Due** *11:59pm Sunday 25 May 2025 (Week 11)*
- **Contributes** 50% to your Assignment 2 result.
- **Group of 3-4 students**.

The Part B of Assignment 2 (2B) requires you to work with your group to implement ML algorithms to train ML models for traffic prediction and integrate the traffic predictor with Part A to develop a fully functioned TBRGS.

In Part B of Assignment 2, the team can use any machine learning technique or combinations of them. You should take advantage of existing libraries such as PyTorch, tensorflow, Keras, Theano, etc. We will provide you with a small dataset from VicRoads for the city of Boroondara that contains only the traffic flow data (the number of cars passing an intersection every 15 minutes). You should use this dataset for the following purposes: training ML models for predicting traffic conditions and testing/evaluating the performance of your models. At the very least, your TBRGS system will need to include the two basic deep learning techniques LSTM and GRU. Your TBRGS system will also need to implement at least one other technique (to be identified by you and approved by your tutor) to train another machine learning (ML) model and give a comprehensive comparison between different models. Your program will at least need to implement the following features:

- At the minimum, the TBRGS will have to be able to train ML models using the Boroondara dataset and give meaningful predictions based on these models
- A GUI will be available for the user input, parameter settings and visualisation (and a configuration file for the defaults).

The Traffic-based Route Guidance Problem

In this problem, there are four main tasks for the team:

1. Implement data processing program to extract data from the given dataset and store it in appropriate data structures to enable ML models to be trained/tested;
2. Implement ML algorithms to train ML models for traffic flow prediction using the provided dataset.
3. Implement a travel time estimation for each edge on the map of the Boroondara area.
4. Integrate Part A of Assignment 2 with the TBRGS to replace the nodes by the intersections, the cost of each edge by the travel time and perform the calculation to find the top-k paths to travel from O to D for any given pair (O,D) of origin and destination.

System requirements

For this assignment, we expect that the team will implement data processing methods and various ML algorithms (including LSTM and GRU) to train ML models for traffic prediction. The team will need to use the given dataset for training and also testing the ML models to evaluate their performance.

The team will also need to port the programs you developed previously for Assignment 2A to enable it to search for optimal paths on the Boroondara map. The edge cost will need to be replaced by the predicted travel time and subsequently, the optimal paths can be calculated and returned.

The basic version of TBRGS will be for the Boroondara area. A user can specify the origin and destination of their trip as the SCATS site number (e.g. origin O = 2000 [intersection WARRIGAL_RD/TOORAK_RD] and destination D = 3002 [intersection DENMARK_ST/BARKERS_RD]). The system then returns up to five (5) routes from O to D with the estimated travel time along each route. To simplify the calculation, you can make a number of assumptions: (i) The speed limit on every link will be the same and set at 60km/h; (ii) the travel time from a SCATS site A to a SCATS site B can be approximated by a simple expression based on the accumulated volume per hour at the SCATS site B and the distance between A and B (We will provide a simplified way to convert from traffic flow to travel time; See the document **Traffic Flow to Travel Time Conversion v1.0.PDF** on Canvas); and (iii) there is an average delay of 30 seconds to pass each controlled intersection. Note that, the objective is not to better Google Maps but to utilise the AI techniques you have

learned (e.g., machine learning for forecasting traffic volume, graph-based search for optimal paths) to solve a real-world problem.

Report file

You must also include a report which has to be either in Microsoft Word or in PDF whose name is your Team ID (for example, 123.PDF) containing your report. The aim of this report is for you to summarise your understanding of the problem and the algorithms you have used to solve the problem. We are most interested in the insights you have gained, especially those **using data obtained by running your software**.

Report Details: The report must be between 8 and 10 pages (excluding cover page and TOC).

- **Cover page:** including details of all group members (i.e., full names and student IDs), signed by all members of the group.
- **Table of contents (TOC)**.
- **Instructions:** Basic instructions of how to use your program. You can also include a **note** containing anything else you want to tell the marker, such as how to use the GUI version of your program, and something particular about your implementation.
- **Introduction:** Introduce the *Traffic Prediction and Traffic-based Route Guidance Problem* and concise information about the algorithms the team have implemented.
- **Features/Bugs/Missing:** Include a list of the features you have implemented. Clearly state if a required feature has not been implemented. Failure to do this will result in penalties. Include a list of any known bugs. Also, anything else you want to tell the marker, such as how to use the GUI version of your program, and something particular about your implementation.
- **Testing:** Provide an overview of the test cases you have created to test your program (either manually or automatically). Report the results of testing your program. By testing, we refer to both the evaluation of the ML models and the software testing you have conducted to ensure that the software works well.
- **Insights:** Present and discuss the qualities of different ML algorithms used in your assignment. Which algorithms are better and why? **Use data collected to support your points.**
- **Research (if applicable):** If you managed to do some additional research to improve the program in some ways, please report it here.
- **Conclusion:** Conclude with a discussion about ML algorithms and their suitability to the problem and about the integration of different modules to produce the end-to-end TBRGS. Include thoughts about how you could improve performance.
- **Acknowledgements/Resources:** Include in your report a list of the resources you have used to create your work. A simple list of URL's is not enough. Include with each entry a basic description of how the person or website assisted you in your work.
- **References:** Cite the sources you referred to in your Assignment (implementation, report, etc.)

Tips:

- All figures and tables need to be properly captioned with sensible descriptions.
- Report presentation should include header/footer information (pages numbers, etc.)

Marking Scheme:

Requirements (or equivalent sections)	***Mark***
Implement appropriate data processing programs to extract data from the provided dataset and store data in appropriate data structures to enable ML models to be trained & tested	**9**
Implement at least 3 ML algorithms (including LSTM and GRU and an algorithm of your choice) for traffic flow prediction.	**7(x3)**
Conduct comprehensive evaluation to compare different ML models	**15**
Integration of Part A & Part B to develop an end-to-end software for TBRGS	**15**
Testing: At least 10 test cases have been created to cover different problem scenarios. Test results have been checked and documented.	**10**

Report: Clear and provide sufficient information about the problem and your algorithm/solution AND the insights you have obtained through your program.	**15**
Research: If you show some initiatives in researching about the problem and solutions, or carrying out extensive tests to provide interesting data about the algorithms, or getting some clever optimization, etc. with a well-written **Research** section in the report to discuss and demonstrate these initiatives	**15**
Total	**100**
You need to follow good programming practice (e.g., well-designed, well-structure codes with clear and helpful comments). Failure to do so get penalty.	**Up to -10**

For each individual student: (the maximum mark a student can get for Assignment 2B is 100 and the minimum mark is 0)	***Mark***
Failure to provide in class updates of the assignment progress to your tutor.	**Up to -40**
Failure to contribute to the work performed by the team or to collaborate with other team members to achieve the desired outcomes for the project.	**Up to -80**

Some ideas for research initiatives

- The standard version of the system only deals with a small area of Melbourne (the city of Boroondara) and uses a very small dataset (traffic volume data for the month of October 2006). Under this research initiative, you can look at more comprehensive datasets from VicRoads for the whole Victoria in multiple years. Data processing for large amount of data will be a challenge.

- The provided dataset may not provide you with details you need. You may want to consider other sources of open data. A good dataset can be found here: https://discover.data.vic.gov.au/dataset/traffic-signal-volume-data
  Some other popular data sources are: https://github.com/mas-dse-c6sander/DSE_Cohort2_Traffic_Capstone/wiki/PeMS-Data-Information and https://www.data.gov.uk/dataset/9562c512-4a0b-45ee-b6ad-afc0f99b841f/highways-england-network-journey-time-and-traffic-flow-data. Under this research initiative, you can choose one of these datasets or a good data source you have access to (and get your tutor's approval) and extend this project to deal with the selected network and dataset.

- Visualising your system predictions and route recommendations. Inspired by Google Maps Traffic, can you build something similar using open source resources such as OpenStreetMap?
  Note: *For some reason, the Latitude and Longitude of the SCATS sites do not map correctly to the actual intersections on Google Maps, you'll have to make adjustments to overcome this issue.*

- For other ideas, please feel free to do the research yourselves and discuss with your tutors to get their approval first before doing the research component.

Part C – Ethical considerations when designing AI systems

- **Due** *11:59pm Sunday 1 June 2025 (Week 12)*
- **Contributes** 10% to your Assignment 2 result.
- **Individual**.

Part C of Assignment 2 (2C) requires you to work individually to explore ethical considerations when making technical and design decisions while developing the TBRGS.

In Part C of Assignment 2, you will work individually. During your participation and involvement in the development of the TBRGS you have encountered a number of design decisions and technical choices that can potentially have some ethical impacts on the product and its users. Your task in Part C (2C) is to identify such design and technical decisions and their potential ethical implications. By documenting and performing an analysis into these decisions and their ethical implications, you can help the development team have a deeper understanding into these issues and see the alternatives to make informed decisions on the design and implementation of the product.

For this document, you will consider the TBRGS system developed by your team and a potential TBRGS+ system that can be developed for the intended functions. For instance, TBRGS+ can be an app run on mobile devices (similar to Google Maps) that can collect personal information such as their locations and even their user profiles (including their personal preferences).

Objectives of the Assignment

On successful completion of Part C, you will be able to demonstrate your ability to apply ethical considerations when designing and developing an AI system. More specifically, your report will:

1. *Identify the key issues related to an AI system you and your team are developing as well as the advanced product that you may design if you are provided with sufficient time and resources*; that is, you are able to identify and describe the design decisions and their potential ethical implications for this product and its possible extensions.
2. *Demonstrate the ability to analyse relevant aspects of these decisions as well as their pros and cons*; that is, you are able to describe the alternatives for each decisions and the impacts they would have on the users and the product's owner and developer.
3. *Provide clear conclusions and recommendations*; that is, you can present your analysis and arguments in a convincing manner to ensure that clear conclusions and perhaps recommendation can be made.

Submission instructions

Your report has to be either in Microsoft Word or in PDF whose name is your Student ID (for example, 1234567.PDF). You must submit your work via Canvas.

Standard late penalties apply - 10% for each day late, more than 5 days late is 0%.
You can follow the following structure for the final report:

- **Cover page:** including your student details (i.e., your full name and student ID).
- **Table of contents (TOC)**.
- **Executive Summary:** (2 paragraphs) First paragraph provides a summary of the potential ethical issues associated with TBRGS or TBRGS+. Second paragraph summarises your analyses and conclusions/recommendations.
- **Main report:** (*1,500-2,000 words*)
  - **Introduction:** Provide a discussion on the design decisions for TBRGS and TBRGS+ and the potential ethical issues associated with these decisions.
  - **Findings:** Describe the alternatives for each decisions and the impacts they would have on the users and the product's owner and developer.
  - **Conclusion and recommendations:** Give a summary of the analyses presented earlier in your report to allow you to arrive at the conclusions and recommendations you want to make.

- **Acknowledgements/Resources:** Include in your report a list of the resources you have used to create your work. A simple list of URL's is not enough. Include with each entry a basic description of how the person/AI tool or website assisted you in your work.
- **References:** Cite the sources you referred to in your report.

Tips:

- If you choose to include a picture/diagram, one figure is equivalent to 50 words.
- Report presentation should include header/footer information (pages numbers, etc.)

Marking Scheme

Requirements (We adapt the **Research Paper Rubric** from **Cornell College** for a more specific **rubric** for this report. See next page.)	***Mark***
Content	**75**
Presentation	**25**
Total	**100**

Adapted from Cornell College's Research Paper Rubric
(https://www.cornellcollege.edu/library/faculty/focusing-on-assignments/tools-for-assessment/ResearchPaperRubric.pdf)

	EXPERT	PROFICIENT	APPRENTICE	NOVICE
TOPIC COVERAGE	The report covers at least two design decisions connecting to both Part A and Part B (for TBRGS) and also at least two additional design decisions related to TBRGS+. The report demonstrates comprehensive understanding of these decisions (including the alternatives and their ethical implications among other things)	The report covers at least two design decisions connecting to both Part A and Part B (for TBRGS) and also at least one additional design decisions related to TBRGS+. The report demonstrates a decent understanding of these decisions (including the alternatives and their ethical implications among other things)	The report covers at least two design decisions connecting to both Part A and Part B (for TBRGS). The report demonstrates a decent understanding of these decisions (including the alternatives and their ethical implications among other things)	The paper does not demonstrate that the author has fully understood and identified appropriate design decisions and/or their alternatives and/or the ethical implications of different alternatives .
DEPTH OF DISCUSSION	In-depth discussion & elaboration in all sections of the paper.	In-depth discussion & elaboration in most sections of the paper.	The writer has omitted pertinent content or content runs-on excessively. Quotations from others outweigh the writer's own ideas excessively.	Cursory discussion in all the sections of the paper or brief discussion in only a few sections.
COHESIVENESS	Ties together information from all sources. Paper flows from one issue to the next without the need for headings. Author's writing demonstrates an understanding of the relationship among material obtained from all sources.	For the most part, ties together information from all sources. Paper flows with only some disjointedness. Author's writing demonstrates an understanding of the relationship among material obtained from all sources.	Sometimes ties together information from all sources. Paper does not flow - disjointedness is apparent. Author's writing does not demonstrate an understanding of the relationship among material obtained from all sources.	Does not tie together information. Paper does not flow and appears to be created from disparate issues. Headings are necessary to link concepts. Writing does not demonstrate understanding any relationships
SPELLING & GRAMMAR	No spelling &/or grammar mistakes.	Minimal spelling &/or grammar mistakes.	Noticeable spelling & grammar mistakes.	Unacceptable number of spelling and/or grammar mistakes.
SOURCES	More than 5 current sources, of which at least 3 are peer-review journal articles or scholarly books. Sources include both general background sources and specialized sources. Special-interest sources and popular literature are acknowledged as such if they are cited. All web sites utilized are authoritative.	5 current sources, of which at least 2 are peer-review journal articles or scholarly books. All web sites utilized are authoritative.	Fewer than 5 current sources, or fewer than 2 of 5 are peer-reviewed journal articles or scholarly books. All web sites utilized are credible.	Fewer than 5 current sources, or fewer than 2 of 5 are peer-reviewed journal articles or scholarly books. Not all web sites utilized are credible, and/or sources are not current.
CITATIONS	Cites all data obtained from other sources. APA citation style is used in both text and bibliography.	Cites most data obtained from other sources. APA citation style is used in both text and bibliography.	Cites some data obtained from other sources. Citation style is either inconsistent or incorrect.	Does not cite sources.

Content = TOPIC COVERAGE + DEPTH OF DISCUSSION + COHESIVENESS
Presentation = SPELLING & GRAMMAR + SOURCES + CITATIONS