**Self-Assessment:** Please highlight where you think your report grade should be. Example below.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | Write simple algorithms using appropriate discrete dat structures to solve computational problems (LO3) | | | | | a | | Use appropriate methods to analyze the efficiency and correctness of algorithms (LO4) | | | | | | |
| **Weight** | **25%** | | | | |  | | **25%** | | | | | | |
| **0 – 29%** | The algorithm does not solve an appropriate problem or has serious errors. There is little or no discussion of how the algorithm works.  No discrete data structure has been used, or the choice of dat structure is inappropriate. | | | | | a | | The analysis is limited and seriously flawed. | | | | | | |
| **30 – 39%** | The algorithm solves part of an appropriate problem. There may be substantial aspects of the problem which are not attempted or explained, or errors in the solution.  The explanation is unclear or missing important details abou how the algorithm works. | | | | | t | | An attempt has been made to analyze an algorithm, but appropriate methods of analysis were not used, and the results of the analysis may be incorrect or meaningless. | | | | | | |
| **40 – 49%** | A rudimentary algorithm solving a basic problem. There ma be some errors which could be corrected with further work. There is a limited discussion of how the algorithm works.  The choice of data structure is inappropriate, or unjustified. | | | | | y | | An attempt has been made to measure the running time of the algorithm for some inputs, but the methodology is unclear, or the measurement may be inaccurate. There is a limited discussion of some other issues relating to efficiency. Analysis of the algorithm’s correctness is vague, or not attempted. | | | | | | |
| **50 – 59%** | The algorithm solves an appropriate problem, though it may have minor errors or fail to account for special cases. There an explanation of how the algorithm works.  The choice of data structure may be inappropriate or poorly justified. | | | | | is | | The running time of the algorithm has been measured accurately for an appropriate range of inputs, and the methodology has been explained. There is some discussion of other issues relating to efficiency.  There is a basic or informal analysis of the algorithm’s correctness. | | | | | | |
| **60 – 69%** | The algorithm correctly solves an appropriate problem. Ther is a clear explanation of how the algorithm works.  At least one appropriate data structure has been used, and th choice has been adequately justified. | | | | | e e | | The efficiency of the algorithm has been accurately measured using an appropriate methodology, which has been explained.  The measurements may include more than one metric.  There is an analysis of the algorithm’s correctness, which may specify pre- and post-conditions for part of the algorithm. | | | | | | |
| **70 – 79%** |  | The algorithm correctly solves a challenging problem. There | | | |  | |  | The efficiency of the algorithm has been accurately measured | | | |  | |
| is a clear explanation of how the algorithm works, and the | | |  | using an appropriate methodology, with multiple metrics and a | | | | |  |
| explanation makes clear references to the relevant parts of the | | | |  |  | clear explanation. The asymptotic complexity of the algorithm | | | | |
| source code. |  | | |  | is given. The efficiency may be compared with appropriate | | |  | |
| Appropriate data structures have been used, and justification | | | | alternative algorithm(s). | |  |
| is given for each with reference to the specific problem. | |  | | There is a formal analysis of the correctness of at least part of | | | |  |
|  | | the algorithm. |  | | |
| **80 - 90%** | A well-designed algorithm which correctly solves a challenging problem. There is a clear, detailed explanation o how the algorithm works, with clear references to the relevant parts of the source code.  Appropriate data structures have been used, and justification is given for each with reference to the specific problem. | | | | | f | | The efficiency of the algorithm has been accurately measured using an appropriate methodology, with multiple metrics and a clear, detailed explanation. The asymptotic complexity of the algorithm is given. The efficiency has been compared with appropriate alternative algorithm(s).  There is a detailed formal analysis of the correctness of the algorithm. | | | | | | |
| **90 –**  **100%** | An excellent algorithm written, explained and evaluated to the highest standards. | | | | |  | | An excellent analysis of the efficiency, complexity and correctness of an algorithm, conducted and explained to the highest standards. | | | | | | |

Global Peace Index 2023: Analyzing Top N Countries Based on Peacefulness Scores and Correlating Socio-economic and Political Factors





**Course title:** Data Structure and Algorithms Module

**Course code:** CMP4272

**Submitted By:**

Diya Kharel

Id: 24152363

**Submitted to:**

Abdul Hamal

Module Leader

**Submission date**: 16th October, 2024

Table of Contents:

1. Introduction: .......................................................................................................................................... 6
   1. Understanding the Global Peace Index and Its Key Factors ........................................................6
   2. Objective of the report................................................................................................................. 6
2. Theory: .................................................................................................................................................. 7
   1. Idea of Selecting Global Peace Index and Socio-economic Factors ........................................... 7
   2. Data Structure and Algorithm used for Ranking Countries .........................................................7
3. Implementation: .................................................................................................................................... 8
   1. Proposed Algorithm to Solve the Problem .................................................................................. 8
      1. Reading Data from CSV file: .................................................................................................. 9
      2. Filtering the countries within a certain Peace Index: .............................................................. 9
      3. Calculating Correlation: ………………………….................................................................10
      4. Use of Quicksort for Ranking: .............................................................................................. 10
   2. Alternative Problem Solution: ................................................................................................... 11
4. Performance Evaluation: ..................................................................................................................... 12
   1. Time Complexity of the Problem: ............................................................................................. 12
      1. Time Complexity in Different Metric: .................................................................................. 12
      2. Comparison of Quicksort with other algorithms: .................................................................. 13
   2. Space Complexity: .....................................................................................................................13
   3. Assertion Table: ........................................................................................................................ 13
5. Conclusion: ......................................................................................................................................... 15
6. References ........................................................................................................................................... 15

Appendix A: Code of the system ............................................................................................................ 16

Appendix B: Output and Data Sample .................................................................................................... 21

Acknowledgement:

I would like to extend my sincere and unabashed thanks to Sunway College and BCU for providing such an opportunity to me to make a technical report presentation on the topic "Global Peace Index 2023: Analysis of Top N Countries Based on Peacefulness Scores and Correlation between some Socio-economic and Political Factors" for the module "Data Structure and Algorithms." Serious challenges were met during the course of studying this module but were solved swiftly with the help and guidance of my instructors. I extend my heartfelt gratitude for their assistance. Besides, the online resources I used benefited recovering the research and completion of this coursework.

Executive Summary:

This report investigates and analyzes the Global Peace Index (GPI) of 2023 in ranking the most peaceful top N governments and socio-economic and political parameters that correlate with the peacefulness scores of these countries. The GPI, developed by the Institute for Economics and Peace, utilizes a variety of peace indicators to rate nations based on levels of safety and security, ongoing conflict, and militarization. Peacefulness scores, which have been determined by analyzing these three key factors, provide an overview of a country's peaceful state.

The report used efficient language data structures (like lists, dictionaries, and objects) and advanced algorithms (including Quick Sort and list comprehension) to process, filter, and rank countries based on GPI scores. These findings show that peace correlates positively with GDP per capita, education, and political stability while emphasizing how safety, conflict reduction, and military de-escalation foster peaceful societies.

The report provides insights of great value to policymakers on the relationships between these factors and peace scores; having combined the advanced algorithms allows processing to take place, which offers a thorough and computationally efficient analysis.

***Keywords - Global Peace Index, Peacefulness Scores, Militarization, Ongoing Conflict, Safety, Socio-economic Factors, Political Stability, Data Analysis.***

1. Introduction:
   1. Understanding the Global Peace Index and Its Key Factors

According to the Global Peace Index (GPI), developed by the Institute for Economics and Peace (IEP), all countries in the world are ranked according to how peaceful they are by cashing in on scores from indicators reflecting the current status of peace within the countries (Institute for Economics and Peace, 2023). The GPI uses criteria such as levels of violence, conflict, incarceration rates, and militarization, yielding a comprehensive picture of the dynamics of global peace (Institute for Economics and Peace, 2023).

This is very important for policy implementers and scholars that will attempt at resolving conflicts and attaining peace.

* 1. Objective of the report

The major aim of this report is to study, in the main, the top N countries according to GPI scores in 2023 while examining how these scores correlate with socio-economic and political factors. This analysis attempts to build a hierarchy of peacefulness among countries and isolate the best practices and effective strategies that lead to an increase in peace-able life (Handayani et al., 2022). The report also aims to see how certain socio-economic indicators like GDP per capita, education, income inequality, and political stability affect peace in the countries concerned (Carlsen, 2017).

The other objective of this report will be to achieve a commensurate level of computational complexity and quality for the ranking results. Finally, we hope to attest to the correctness of the proposed approach via the application of an assertion table. This comprehensive approach will also provide insightful information to policymakers in developing and implementing strategies that promote peace through socio-economic development.

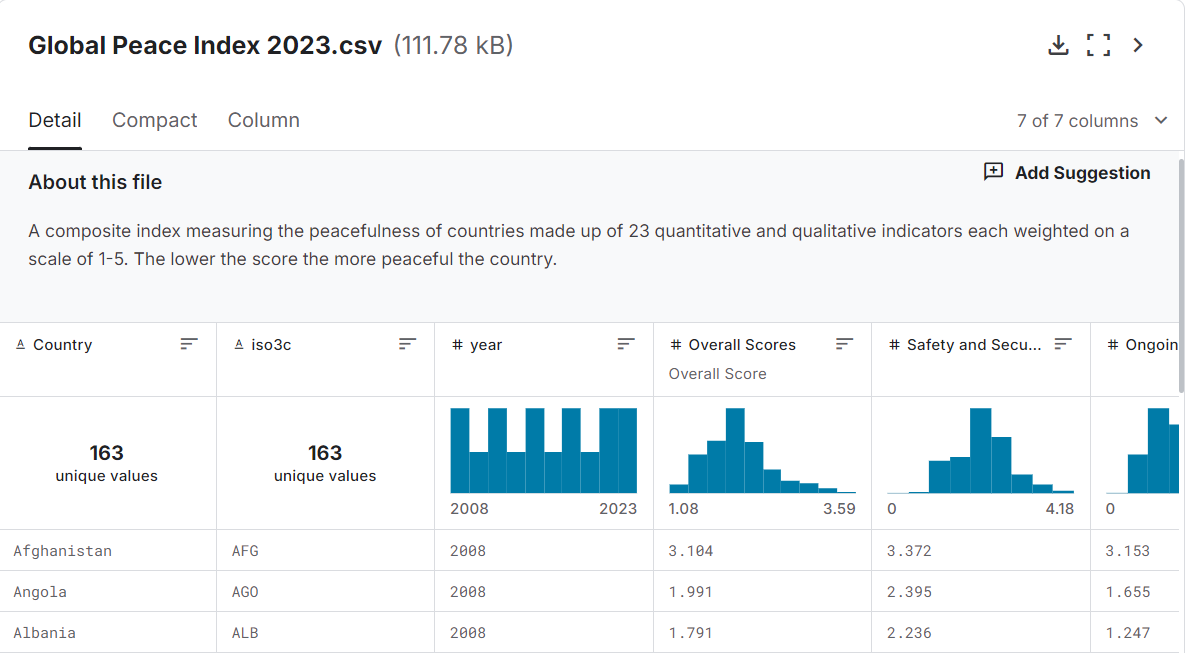


Figure.1. Global Peace Index file in kaggle

1. Theory:
   1. Idea of Selecting Global Peace Index and it’s Factors

Based on the selection of GPI as a primary index, broader analyses of national peace are carried out. The correlations of GPI scores with various socio-economic indicators, including GDP per capita, levels of education, and political stability, among others, will be established in this report (Shrestha, 2024). These relationships, when analyzed, may give a better understanding of how improvements in socio-economic conditions relate to better national peace.

Countries with higher wealth are generally said to be most likely to have greater investments in public services that can promote stability and reduce the levels of violence (Handayani et al., 2022). It was also noted that the level of education impacts greatly on crime rates and civic engagement; as education improves, crime rates drop while civic engagement rises (Carlsen, 2017).

* 1. Data Structure and Algorithm used for Ranking Countries

This section gives a quick summary of the algorithms and data structures, respectively, required for the proposed algorithm.

2.2.1. Data Structure:

1. **Class**: Classes are also data structures in Python, but they are more advanced than basic dictionaries, tuples, and lists (Mailund, 2021). It consists of all the socio-economic attributes of the dataset. Using a class makes it easy to represent each country and is reusable. Ultimately it provides a clear structure and to increases the readability of the code.

1. **List**: In a list data structure, the country class instance stores and manipulates this data. The operations involve sorting and filtering. They are easy and efficient to store and manipulate sequences of data. They allow random access as well as provide easy traversal for reading.

1. **Dictionary**: To efficiently store and manipulate the data, a list data structure is used here as an instance of the “country” class. It is particularly useful for storing and accessing correlation coefficients, where each socio-economic factor in the data is a key, and its correlation with the happiness score is the value. Thus, allowing easy and efficient retrieval of correlation data for analysis and visualization.

2.2.2. Algorithm:

1. **Iteration and Collection:** This algorithm goes through a CSV file with GPI data and creates a Country object for each row. It reads the file line by line, which is efficient for handling large datasets. Each row is transformed into a suitable data structure that includes important attributes like the country name, GPI score, GDP per capita, education level, and various socio-economic indicators. This method improves data organization and makes it easier to perform operations such as filtering and sorting later on.

1. **List-Comprehension**: List comprehension in Python serves the purpose of sorting countries with proper GPI scores or a certain level of socio-economic parameters. This is used because they outperform “For” loops in speed due to the better internal architecture of the language, hence the neatness and brevity of codes (Wadler, 1992). Their time complexity is linear O(n), thus suitable for small to average datasets.

1. **Quick Sort:** The reason Quick Sort is used is that the average-case time complexity is O(n log n) and the sorting is done in-place thus fitting large amounts of data. However, the worst-case time complexity of O(n²) is associated with Quick Sort but this stage may be avoided by proper pivot selection, for example, the median or middle element. Quick sort is considered a divide and conquer algorithm, which makes it so efficient for the purpose of sorting in this case.

3. Implementation:

* 1. Proposed Algorithm to Solve the Problem

This section presents a mechanism to analyze and visualize the Global Peace Index (GPI) 2023 data through ‘GPI\_2023.csv’ dataset. The algorithm does well in identifying and analyzing the causes of peace scores for various countries. The code is constructed in an object oriented manner with the Country class having attributes such as rank, country, peace score, gdp per capita, military spending, political risk, unemployment, human rights and government efficiency. These attributes of the Country class are important because they enable further analysis and subsequent visualizations to be carried out.

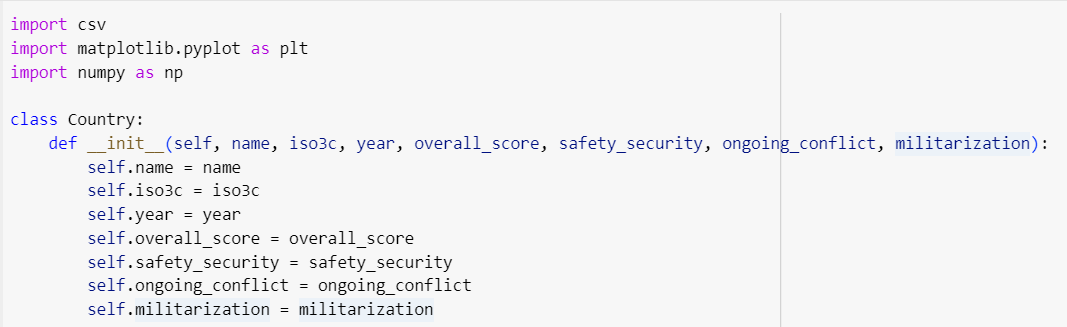


Figure.2. Code snippet of Country Class

* + 1. Reading Data from CSV file:

The read\_data(file\_path) method is implemented in such a way that it reads a CSV file and populates the list with Country type objects. It does so by looping through each row of the CSV file creating the Country objects with the GPI data. After reading them from the file, empty lists which will hold the objects are created.



Figure.3. Code Snippet of Reading CSV file code

* + 1. Filtering the countries Based on Peace Score:

The filter\_by\_score function is invoked when the user wants a specific list of countries according to the peace scoring list. The function performs this operation using a list comprehension in Python containing for loop to consider each country’s score and check whether it is less than the min\_score and max\_score values. If yes country’s peace score is in this range then this scope is encourages taking such cuisines for further investment in the country’s violence score.

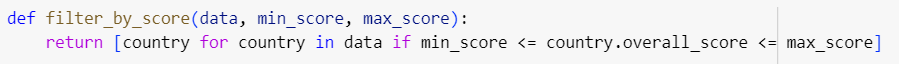


Figure.4. Code Snippet of Filtering Code

* + 1. Calculating Correlation:

The calculus\_correlation function is useful in establishing the relationship of certain social economic and political factors with the overall peace score of a given country. The peace scores of all the countries are put into one list called peace\_scores and using some statistical methods the peace scores are correlated with value aspects such as GDP per capita, military expenditure, political stability among others. These findings are summarized in a table where every aspect is correlated with their correlation.

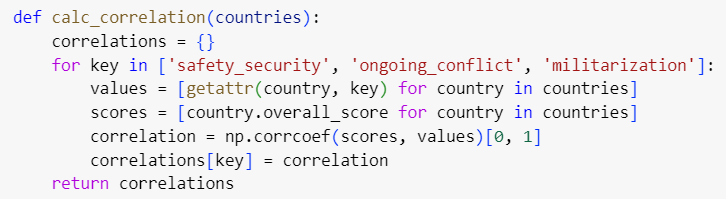


Figure.5. Code Snippet of Calculating Correlation

* + 1. Use of Quicksort for Ranking Countries:

In order to establish a ranking among countries according to their peace scores, a quicksort algorithm is used. This algorithm picks a point in the list of countries (i.e. a pivot) divides the list into three slices- countries having higher peace scores, lower peace scores and equal peace scores relative to this point In History. This process in implemented in a recursive manner for the higher score group and lower score group, sorting within each of the groups. Once all the sorted groups are combined, the sequence of such groups is reversed to rank the countries in descending order of their peace.

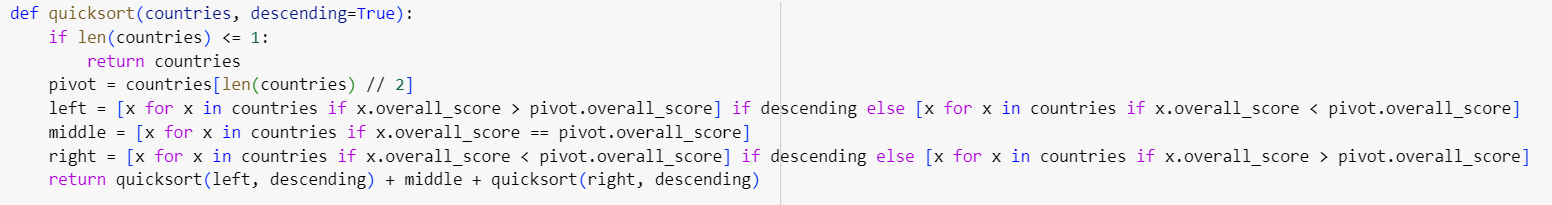


Figure.6. Code Snippet of Use of Quick Sort

* 1. Alternative Problem Solution:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Functionality** | **Current Data Structure** | **Alternative**  **Data**  **Structure** | **Advantages of Current** | **Disadvantages of Current** | **Advantages**  **of**  **Alternative** | **Disadvantages of Alternative** |
| **Storing**  **Country Data** | Class | Dictionary | Encapsulation, reusability, easy to extend. | Overhead of object creation and attribute access. | Simple keyvalue access, flexible structure. | Less strict structure, harder to maintain consistency. |
| **Storing List of Countries** | List | Linked List | Fast random access, simple to iterate and use. | Inefficient for frequent  insertions and deletions. | Efficient insertions and  deletions, dynamic size. | No direct random access, more memory overhead. |
| **Filtering Data** | List  Comprehension | Generator Expression | fast iteration for small datasets. | Entire list needs to be created in  memory, can be inefficient for large datasets. | Memory efficient, elements  generated on-the-fly. | Only allows single-pass  iteration, can  be slower for small datasets. |
| **Correlation Calculation** | Dictionary | Pandas Data Frame | Simple keyvalue storage, easy to access and iterate. | Limited functionality  for statistical  operations and data  manipulation. | Rich functionality  for statistical  operations, easy data  manipulation and  visualization. | More complex to set up and  use, additional dependencies. |
| **Sorting** | Quicksort | Merge Sort | Average case performance  (O(n log n)), in-place sorting  (memory  efficient). | Worst case performance  (O(n²)), not stable. | Stable sorting  (preserves order of equal  elements),  guaranteed  O(n log n) performance. | Requires additional  memory (not in-place). |

Table.1. Alternative Possible Algorithm that could be used

1. Performance Evaluation:
   1. Time Complexity of the Problem:
2. **Reading the Data:**

**Time Complexity:** O(n), where n is the total number of rows in the CSV file. Processing each row once to generate Country object requires O(n) which is the processing of all records in the table.

1. **Filtering the Countries:**

**Time Complexity:** O(n), where n is the number of countries in data. It filters the countries using the score but this is done in one iteration of the list of countries.

1. **Calculating Correlation:**

**Time Complexity:** O(nm), where “n” is the number of countries and “m” is the number of indicators. This is because it calculates correlation for each indicator with respect to the scores.

1. **Quick sort:**

**Time Complexity:** Time Complexity: O(inner n\*m), where n is the number of countries and m is the number of indicators. This is because here correlation is being computed for each indicator based on the scores.

Rather than being satisfied with all the time complexity of our algorithm, we are able to isolate those which are the most troublesome.

* + - * O(n.m) from calculating correlation
      * O(nlogn) from quicksort

Since, O(n,m) dominates O(nlogn), therefore the total time complexity T(n,m) is,

T

* + 1. Time Complexity in Different Metric:

|  |  |  |  |
| --- | --- | --- | --- |
| Function | Best Case Complexity | Worst Case Complexity | Average Case Complexity |
| read\_data(file\_path) | O(N) | O(N) | O(N) |
| filter\_by\_score(data, min\_score, max\_score) | O(1) | O(N) | O(N) |
| calc\_correlation(countries) | O(N) | O(N.M) | O(N.M) |
| quicksort(countries,  descending=True) | O(n \log n) | O(n^2) | O(n \log n) |
| print\_countries(countries, num\_to\_print) | O(1) | O(num\_to\_print) | O(num\_to\_print) |
| print\_correlations(correlations) | O(M) | O(M) | O(M) |

Table.2. Time Complexity Comparison

* + 1. Comparison of Quicksort with other algorithms:

|  |  |  |
| --- | --- | --- |
| Time Complexity | Algorithm | Disadvantages Compared to Quicksort |
| Merge Sort | O(nLogN) | * Typically requires additional space for recursive calls (stack space). * More complex implementation |
| Counting Sort | O(N) | * Limited to a specific range of integer inputs * Requires additional space proportional to the range of input values. |
| Bubble Sort, Selection Sort | O (N.N) | * Quadratic time complexity, slower for large dataset * Simple to implement but less efficient |

Table.3. Comparison of other Algorithms with Quick Sort

After examining the table above we can see that merge sort is a better option in terms of time complexity than quicksort but, only for large datasets. Thus, quicksort is the best.

* 1. Space Complexity:

This is how the space complexity of the code looks like for each function. Read is O(n) without the capacity optimization, selects countries are O(k), and quicksort copy-and-paste sorting takes its due: Both read data and this quicksort model function use extra memory for each country related to n or filtered gives us values.

Thus, the Total Space Complexity = O(N)

* 1. Assertion Table:

|  |  |  |
| --- | --- | --- |
| Assertion | Code Snippet | Functionality |
| Filtering Countries |  | Working |

|  |  |  |
| --- | --- | --- |
| Reading the Data  and storing  in list |  | Working |
| Correlation calculation |  | Working |
| Quick Sort |  | Working |
| Printing the data |  | Working |
| Visualizing Data |  | Working |

Table.4. Assertion Table

1. Conclusion:

Global Peace Index 2023 | TwoHabThe Global Peace Index The relationship between a country's socio-economic and political conditions is hidden behind the peacefulness of this country. It is from these correlations that policymakers of what they can do in order to foster peace through socio-economic development. The results show the need to invest in education, work towards income equality and achieve political normality between nations for world peace.

1. References

Mailund, T. (2021) ‘Data Structures, objects, and classes’, *Introduction to Computational Thinking*, pp. 317–367. doi:10.1007/978-1-4842-7077-6\_11.

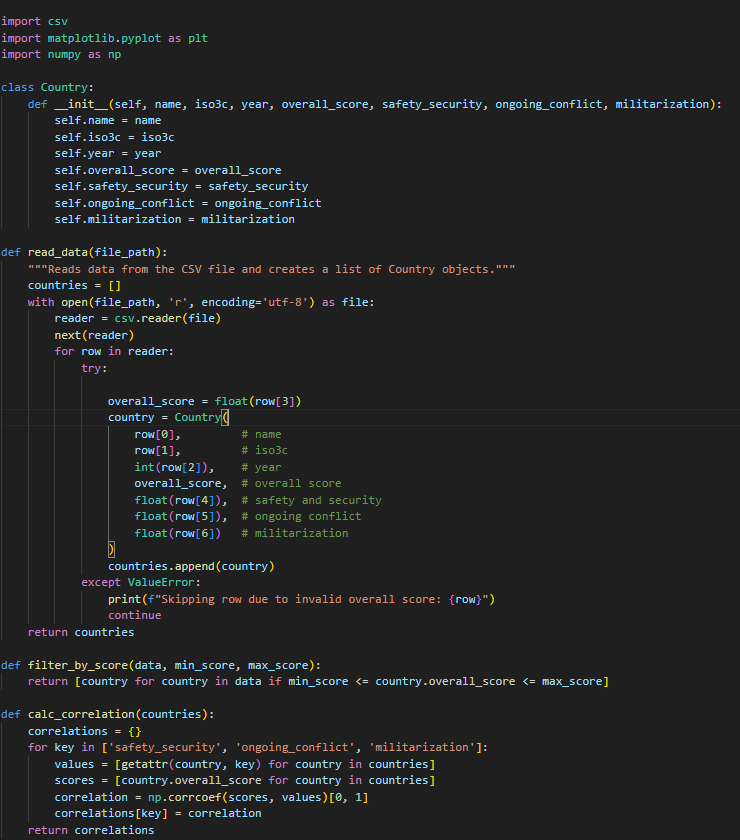
Dosad, D. (2023) *Global Peace Index*, *Kaggle*. Available at:

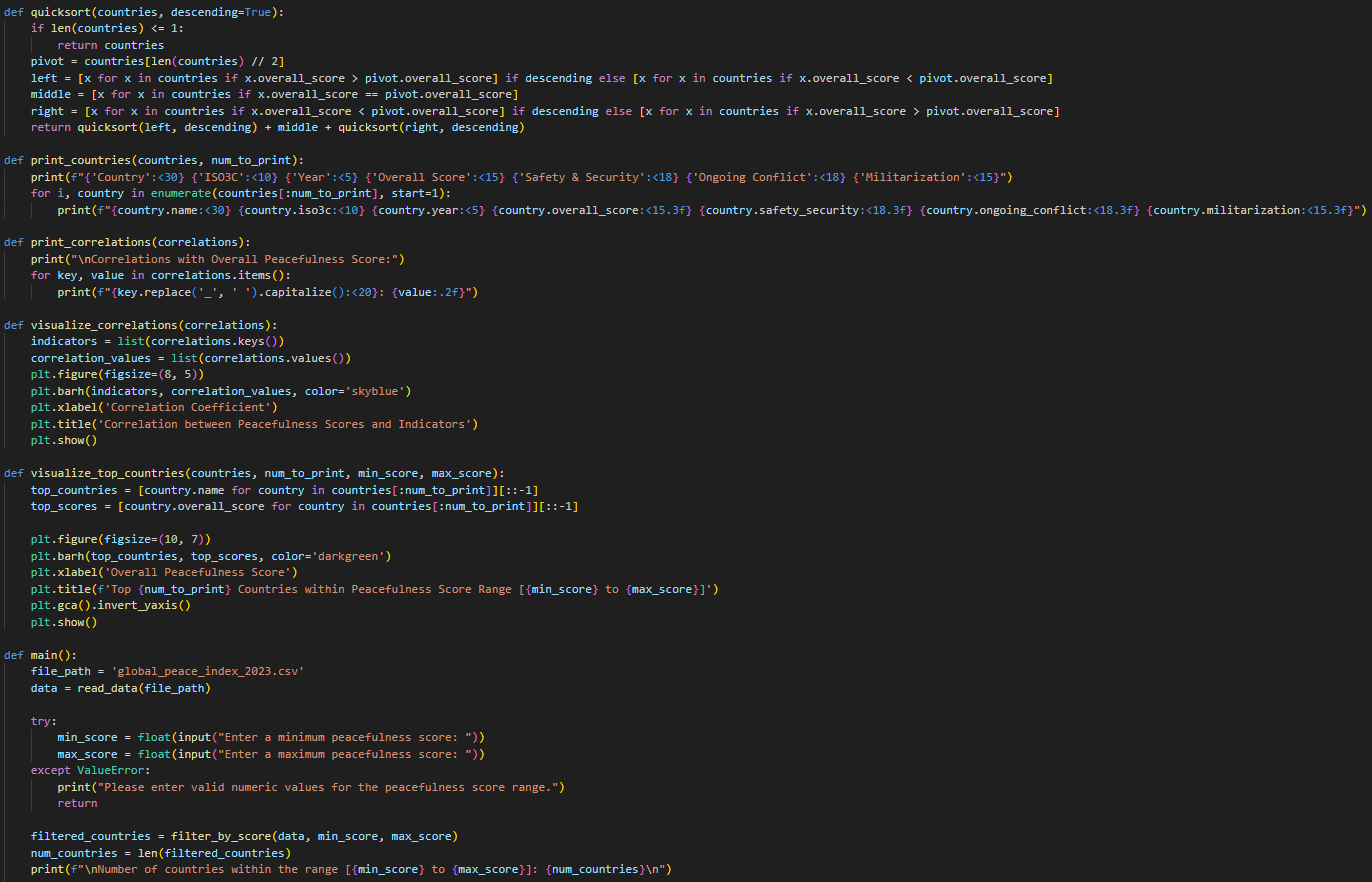
https://www.kaggle.com/datasets/ddosad/global-peace-index-2023?resource=download (Accessed: 17th September, 2024).

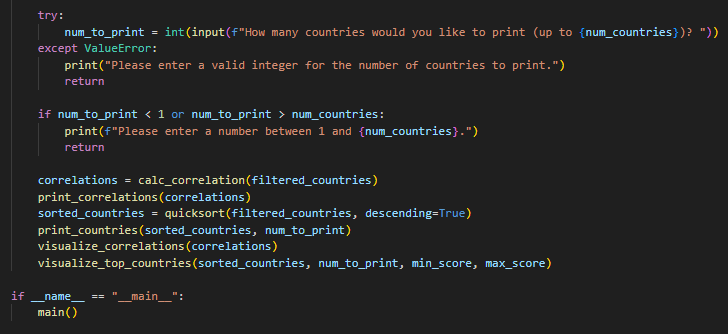
Wadler, P. (1992) ‘Comprehending monads’, *Mathematical Structures in Computer Science*, 2(4), pp. 461–493. doi:10.1017/s0960129500001560.

Institute for Economics and Peace (2023). Global Peace Index 2023.

Appendix A: Code of the system:





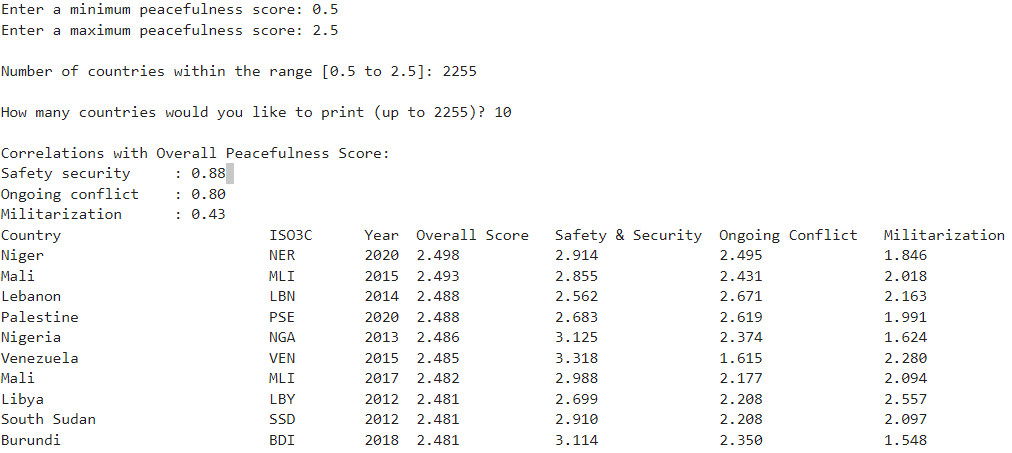


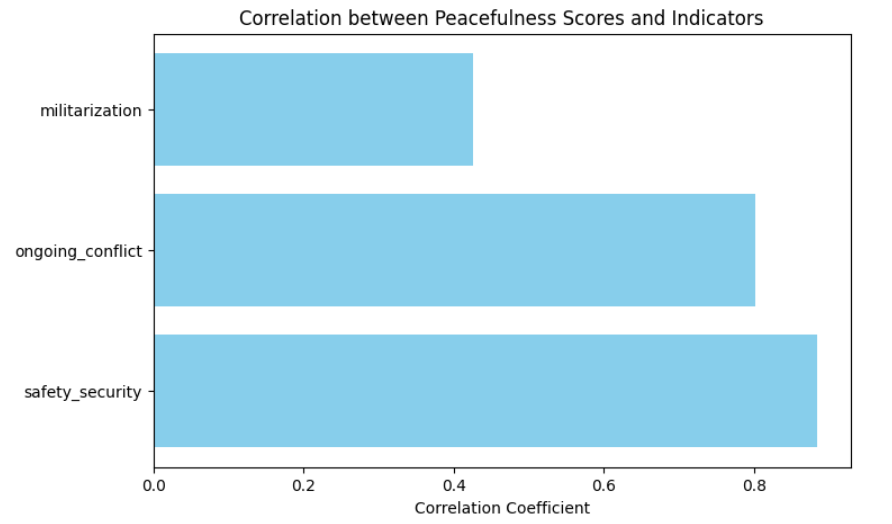
Appendix B: Output and Data Sample

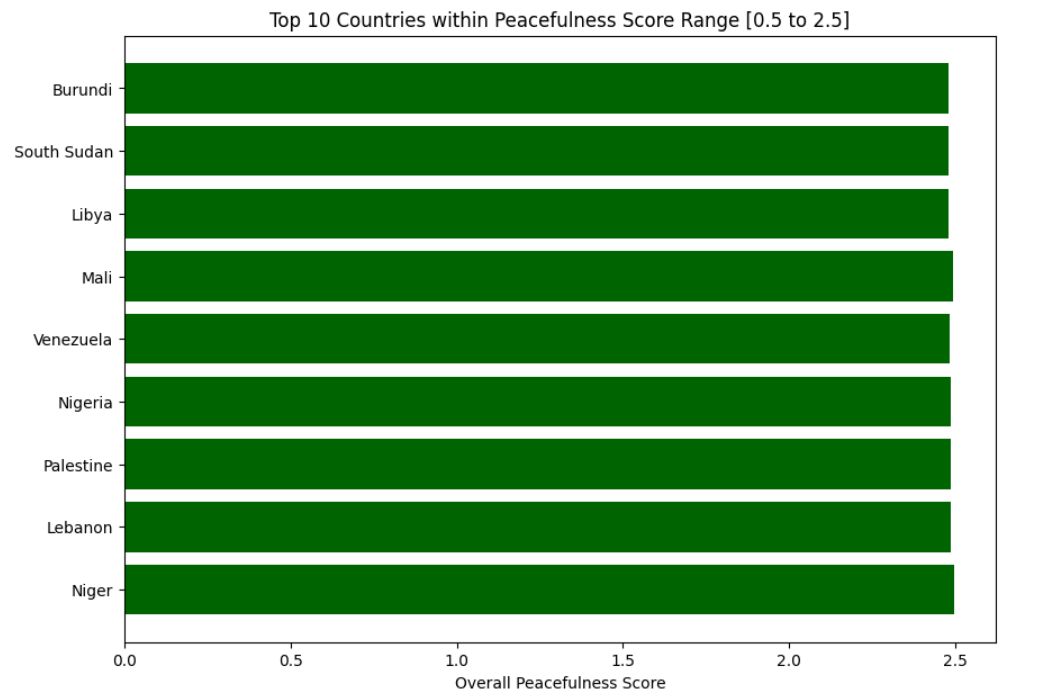
Data Sample: a sample of data used in the project,

|  |  |  |
| --- | --- | --- |
| Rank | Countries | Overall Score |
| 1 | Afghanistan | 3.104 |
| 2 | Angola | 1.991 |
| 3 | Albania | 1.791 |
| … | … | … |
| 2609 | Zimbabwe | 2.3 |

**Output case 1**: For Peacefulness Score of 10 countries with index between 0.5 and 2.5.







**Output case 2**: For Peacefulness Score of 64 countries with index between 2 and 3.

