Laboratory practice No. 2: Brute Force or Exhaustive Search

Miguel Sosa Villegas
Universidad Eafit
Medellín, Colombia
msosav@eafit.edu.co

Sergio A. Córdoba Muriel
Universidad Eafit
Medellín, Colombia
sacordobam@eafit.edu.co

Miguel Jaramillo Arenas
Universidad Eafit
Medellín, Colombia
mjaramil20@eafit.edu.co

3) *Practice for final project defense presentation*

3.1 *We used permutation without repetition, we created all the permutations and turn them into lists, the we checked each list, we checked that the first node was not the final node and if the final node in the list was not the final node we pass to the other list, and so on, if it meets the requirements the it finds if there's any weight between it and the next node, if it does it sums a the aux variable, and so on, then it compares to the peso variable and gives the shortest path.*

3.2 *The complexity is **O(V*E)** in the worst case when all the nodes are connected between them.*

3.3 *It would take about 5,000 seconds which is like an hour and 20 minutes.*

3.4 *The data structures we used were the array of one and two dimensions. The one-dimensional one to represent the possible answers and the two-dimensional one to specify the places where the queens could not be placed. The algorithm works using backtracking, using a two-dimensional array of booleans that indicates in which positions a queen cannot be placed. And in case the position that follows to try is located in a false, then it skips it and continues with the following ones. At the end it returns the number of answers.*

3.5 *The complexity for exercise 2.1 is O(n^2) because there is a for loop inside another for loop.*

3.6 *N is the number of rows that will be entered.*

4) *Practice for midterms*

4.1. Subarray Max

4.1.1. `actual > max`

4.1.2** Complexity in the worst case:* ***O(n^2)

4.2. Sorting

4.2.1 `sort(arr, k);`

4.2.2 *Complexity in the worst case:* ***O(**n**)***

4.3. Text String

4.3.1 *Line number 12:* `if(j == m) return i-m;`

4.3.2 *Line number 13:* `else return n;`

4.3.3 *Complexity in the worst case, in terms of n and m:* ***O(**$n * m$**)***

4.4. Range

4.4.1 *Line number 8:* `int rem = temp%10;`

4.4.2 *Complexity in the worst case:* ***d. O(|N-M|xM)***

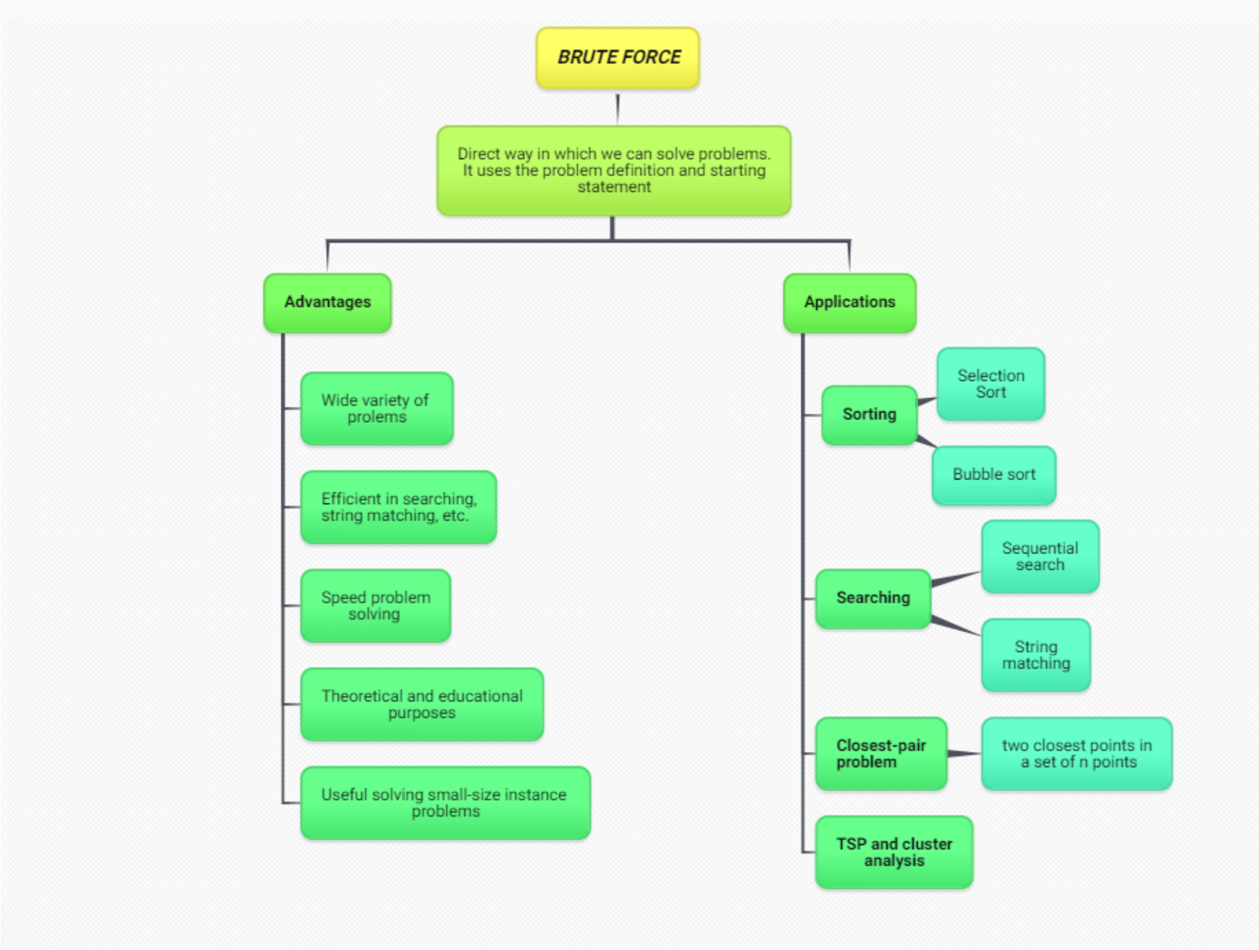
4.5. Array

4.5.1 *Line number 7:* `for(int j = i+1; j < n; j++)`

4.5.2 *Line number 10:* `can = can || left == right;`

5) Recommended reading (optional)

Conceptual Map


BRUTE FORCE
Direct way in which we can solve problems. It uses the problem definition and starting statement
Advantages
Wide variety of prolems
Efficient in searching, string matching, etc.
Speed problem solving
Theoretical and educational purposes
Useful solving small-size instance problems
Applications
Sorting
Selection Sort
Bubble sort
Searching
Sequential search
String matching
Closest-pair problem
two closest points in a set of n points
TSP and cluster analysis

Summary

*Brute force is a direct way in which we can solve problems. It's based on the problem definition where the concepts are involved, and also, the problem starting statement. The **"force"** that was mentioned in this problem solving strategy means the one who comes from computers and not our minds.*

There are several advantages when using brute-force algorithms, such as:

- *Applicable to a wide variety of problems.*
- *Efficiency in searching, matrix multiplication and string matching.*
- *Speed problem solving.*
- *Used in solving small-size instances of a problem.*
- *It can serve both theoretical and educational purposes.*

In addition, brute-force algorithms can be used in a wide field as:

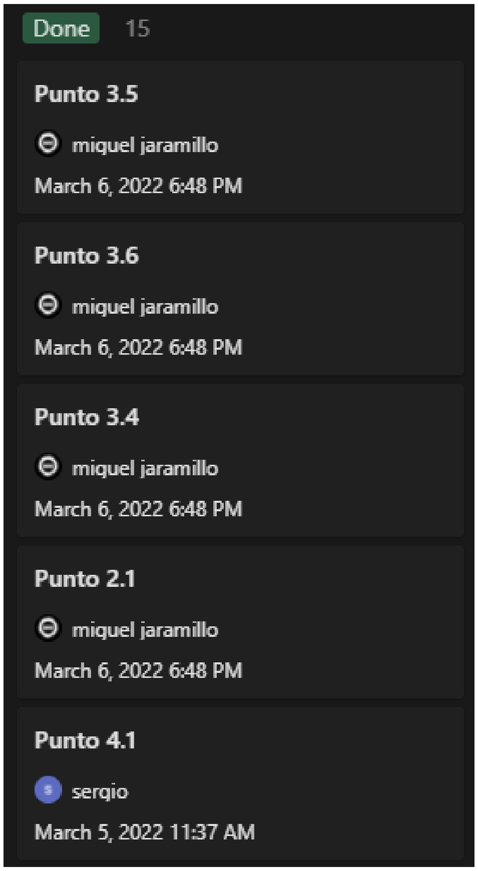
***Sorting:** in which selection sort and bubble sort algorithms are the most outstanding.*

Searching: *in which sequential search and string matching are the most outstanding.*
Closest-pair problem: *the objective is to find the two closest points in a set of* ***n*** *points. From this application follows cluster analysis, where the points are supposed to be an organized hierarchy based on similarity metrics.*
Also, we can highlight solving the convex hull with brute-force algorithms, exhaustive search where it is required to find an element with some properties in a domain that grows faster.

Finally, talking about the ***Traveling Salesman Problem (TSP)*** *we can't leave brute force by a side. It's all about finding the shortest way given a set of n cities, that visits each city only once before returning to the starting point. If we talk about graphs, we could say the cities represent vertices and the distance between them represent the weights.*

6) Teamwork and gradual progress (optional)

6.1 Kanban Board



Done 15

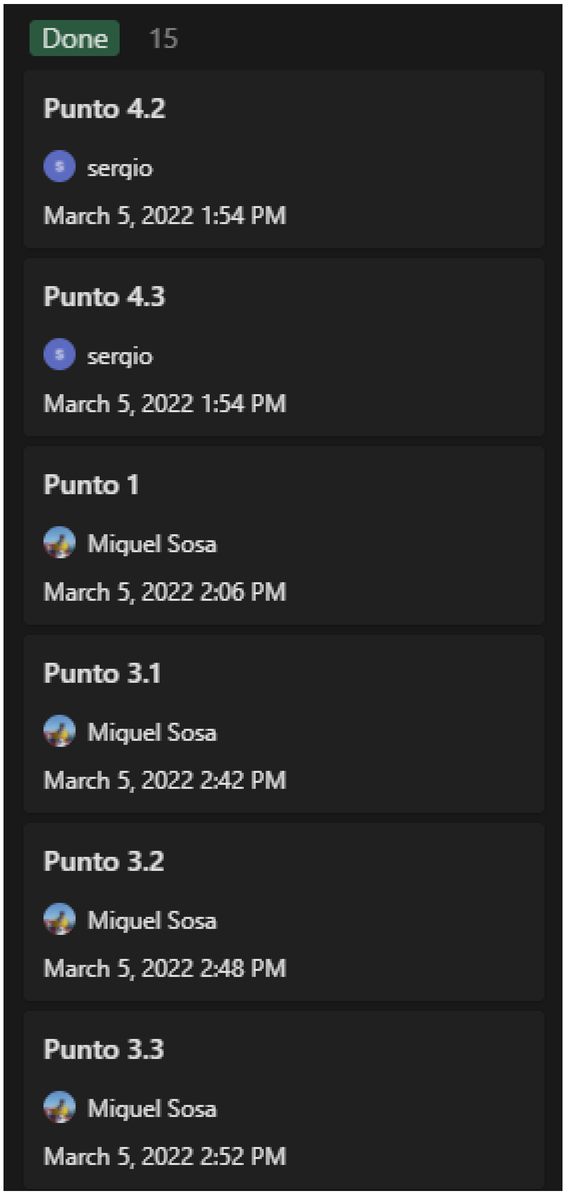
Punto 3.5
⊖ miquel jaramillo
March 6, 2022 6:48 PM

Punto 3.6
⊖ miquel jaramillo
March 6, 2022 6:48 PM

Punto 3.4
⊖ miquel jaramillo
March 6, 2022 6:48 PM

Punto 2.1
⊖ miquel jaramillo
March 6, 2022 6:48 PM

Punto 4.1
S sergio
March 5, 2022 11:37 AM



Done 15

Punto 4.2
sergio
March 5, 2022 1:54 PM

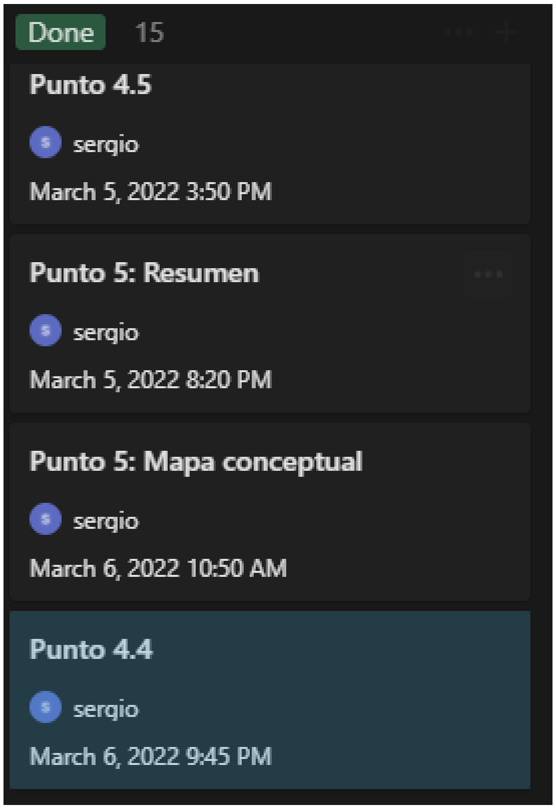
Punto 4.3
sergio
March 5, 2022 1:54 PM

Punto 1
Miguel Sosa
March 5, 2022 2:06 PM

Punto 3.1
Miguel Sosa
March 5, 2022 2:42 PM

Punto 3.2
Miguel Sosa
March 5, 2022 2:48 PM

Punto 3.3
Miguel Sosa
March 5, 2022 2:52 PM



Done 15

Punto 4.5
sergio
March 5, 2022 3:50 PM

Punto 5: Resumen
sergio
March 5, 2022 8:20 PM

Punto 5: Mapa conceptual
sergio
March 6, 2022 10:50 AM

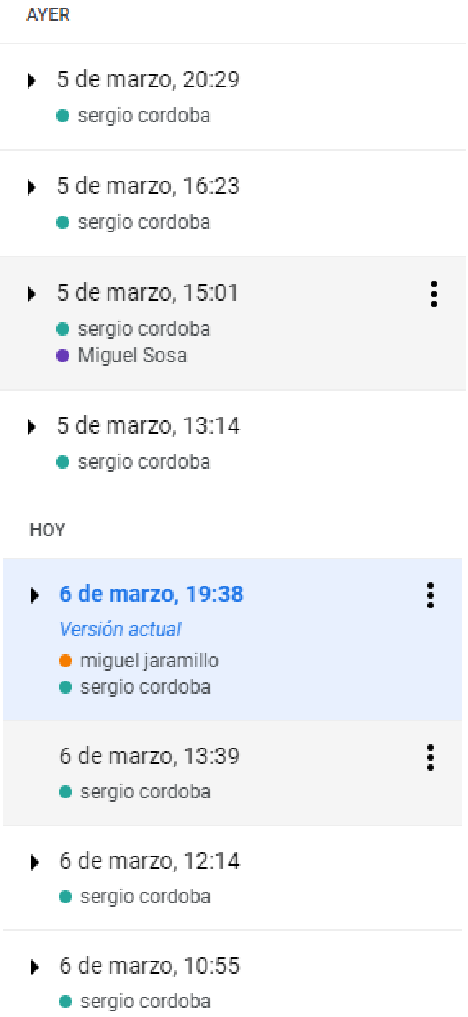
Punto 4.4
sergio
March 6, 2022 9:45 PM

6.2 History of changes of the code

```
commit a6f65d4dc02516dde7797912e82e1a7bfdda446c (HEAD -> master, origin/master, origin/HEAD)
Author: mstermigol <85334763+mstermigol@users.noreply.github.com>
Date:   Sun Mar 6 01:46:41 2022 -0500

    Add files via upload

commit fe6f72e2b58099e505cc0f2dd6221d70428960b9
Author: mstermigol <85334763+mstermigol@users.noreply.github.com>
Date:   Sun Mar 6 01:46:29 2022 -0500

    Delete Punto2.py

commit 82b7882829456ac2420f9aa85084ca6751a5229e
Author: mstermigol <85334763+mstermigol@users.noreply.github.com>
Date:   Sun Mar 6 01:45:02 2022 -0500
```

```
    Add files via upload

commit fdf4d545babda18f98528c0fdf0f4dd3eb11066a
Author: Miguel S <85181687+msosav@users.noreply.github.com>
Date:   Sat Mar 5 19:25:18 2022 -0500

    Add files via upload
```

6.3 History of changes of the report



AYER

5 de marzo, 20:29
● sergio cordoba

5 de marzo, 16:23
● sergio cordoba

5 de marzo, 15:01
● sergio cordoba
● Miguel Sosa

5 de marzo, 13:14
● sergio cordoba

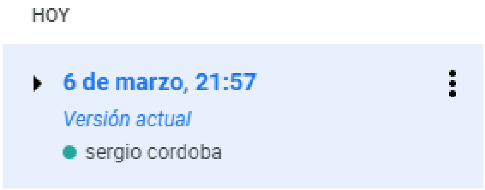
HOY

6 de marzo, 19:38
Versión actual
● miguel jaramillo
● sergio cordoba

6 de marzo, 13:39
● sergio cordoba

6 de marzo, 12:14
● sergio cordoba

6 de marzo, 10:55
● sergio cordoba



HOY

6 de marzo, 21:57

Versión actual

● sergio cordoba