

**DNA of Algorithms**

**BS (Computer Science)**

**Dr. Aarij Mahmood Hussaan**

**Project**

**Instructions**

For this project you are required to solve a real-world problem using an advance algorithm. You can work in a team of maximum 3 members. You can choose an algorithm, from the following list or you can come up with your own proposal of an algorithm. For any algorithm you choose, you also need to propose which **real world problem** you will be solving.

* Traveling sales man problem (TSP)
* Maximum Flow
* Manipulation (insertion, deletion, modification) of B+ Trees
* Algorithms of Linear Programming
* Multithreaded matrix multiplication
* Discrete/Fast Fourier Transform
* The RSA public-key cryptosystem
* Determining whether any pair of segments intersects
* Finding the convex hull
* Finding the closest pair of points
* The vertex-cover problem
* The set-covering problem

For any algorithm that you choose, you have to do the following in the form of a report:

1. **Title page**, which includes the Algorithm name, group members’ names and the date
2. **Problem**. Description of the real-world problem that the algorithm solves
3. **Real Life Examples**. Substantial example of the problem. Give substantial examples. Explain the inputs, for example if the input is a graph then what the vertices and edges represent.
4. **Complexity**. Explain and prove the complexity of the problem
5. **Algorithm**. This is a verbal description
6. **Algorithm Correctness**. This can be a proof or an argument.
7. **Pseudo Code**. Only for the top level of the implementation
8. **Cost analysis**. This should be the complete cost for all the different input cases and data structures used.
9. **Implementation**. Description of classes used and difficulties during implementation.
10. **Correctness Runs**. Description of correctness runs and why you used these cases. This should explain what input is used and why. You may need to explain the expected output.
11. **Correctness Runs Results**. Results of correctness runs. Give the input and output. Make a conclusion.
12. **Cost Runs**. This is a description of the cost run, should explain the different inputs and expected output.
13. **Cost Runs Results**. This section should include graphs, and verbal description for different input types and data structures.
14. **References**. Web resources are permitted
15. **Running Code**. Description of how to run your program including generating test cases. Give the names of the main classes.

All team members have to participate in the development of the project. I will conduct an extensive VIVA before awarding any kind of credit to team members.

A maximum of three teams will be allowed to work on a single class of algorithms, for example, only three teams can implement the Robin Karp’s algorithm for String matching, others can choose other algorithms for String matching.

You have to select at-least three algorithms and send me an email (only one per group). The algorithms will be allotted to you by me on a First Come First Served basis.

**SUBSTANTIAL BONUS:** Implement a GUI for the algorithm. This GUI should allow users to give input in an easy and intuitive manner. The output should be a graphical representation of all the steps taken by the algorithm to solve the problem.

**Submission :- Deadline for submission :** 24 April 2016 23h59 p.s.t

**File format:** java (Java source file)

**Instructions:** All your code should be in a single document.

Follow the following instructions:

1. **Go to turnitin.com**
2. **Click on “Create Account”**
3. **Select “Create a new account” as a Student**
4. **Enter the following class id 12130111**
5. **Enter the following enrollment password dnaoasp16**
6. **Go to project and submit your file when ready**
7. **Only one of the group member should submit the project**
8. **If you already have an account you can skip the first 3 steps.**