# Developer Test

You are tasked with developing a command-line utility that processes data in XML and JSON formats, performs some operations, and outputs the results. The utility should be written in C++ and also include Unix shell scripts for automation.

## Requirements:

1. **Input Data:**

* Accepts both XML and JSON files as input.
* The input files will contain information about employees in a company. Each employee record will have attributes like name, ID, department, and salary.

1. **Functionality:**

* Parse both XML and JSON files to extract employee information.
* Calculate the average salary of all employees.
* Identify the highest paid employee and print their details.
* Sort the employees based on their IDs and output the sorted list.

1. **Output:**

* Display the average salary.
* Print details of the highest paid employee.
* Display the sorted list of employees by ID.

1. **Implementation:**

* Use C++ for implementing the core functionalities.
* Utilize Standard Template Library (STL) for data manipulation where applicable.
* Implement Unix shell scripts to automate the process of running the utility, handling input/output files.

1. **Testing:**

* Provide test cases for both XML and JSON inputs.
* Ensure error handling for incorrect file formats or missing files.

1. **Version Control:**

* Utilize Git for version control.
* Include a README file with instructions on how to compile, run, and test the utility.
* Commit your changes at appropriate intervals with descriptive commit messages.

## Evaluation Criteria:

* Correctness of the implementation: The utility should produce accurate results according to the requirements.
* Code quality: Well-structured, readable, and efficient code written in C++.
* Handling of XML and JSON parsing: Ensure proper handling of both formats with appropriate libraries.
* Usage of STL: Utilize STL containers and algorithms effectively where necessary.
* Shell scripting: Demonstrates proficiency in Unix and shell scripting for automation.

Sample Input:

**Employees.xml:**



**Employees.json:**



## Delivery & conditions

1. Share the output code via a public Git repository (e.g., GitHub, GitLab, Bitbucket).
2. Include a README file with instructions on how to compile, run, and test the utility.
3. Allowed time: **2 days.**
4. **Send your response to:** [**florencio.moralesbermudez@amadeus.com**](mailto:florencio.moralesbermudez@amadeus.com) **and** [**camila.medinacuervo@amadeus.com**](mailto:camila.medinacuervo@amadeus.com)