**System Secutiry Assingment 3**

**Name: Jeslyn Ho Ka Yan**

**Sim id: 10241485**

**Section 1: Initial Input Phase**

**1.1 Data Parsing and Internal Storage**

The program reads data from two input files, Events.txt and Stats.txt, which define the events and their statistical properties. The program parses these files and stores the data in Python dictionaries as follows:

* **Events Dictionary (events):**
  + This dictionary stores the details of each event from Events.txt.
  + Each event is represented as a key-value pair, where the key is the event name, and the value is a dictionary with the attributes:
    - **types, min, max, weight**
* **Statistics Dictionary (stats):**
  + This dictionary stores the statistical properties of each event as defined in Stats.txt.
  + Each event name is a key, and its value is a dictionary with:
    - **Mean and standard deviation**

**It would look like this**

Parsing Events file: Events.txt

Number of events in Events.txt: 5

Events parsed successfully: {

'Logins': {**'type'**: 'D', **'min'**: 0, **'max'**: inf, **'weight'**: 2},

'Time online': {'type': 'C', 'min': 0, 'max': 1440, 'weight': 3},

'Emails sent': {'type': 'D', 'min': 0, 'max': inf, 'weight': 1},

'Emails opened': {'type': 'D', 'min': 0, 'max': inf, 'weight': 1},

'Emails deleted': {'type': 'D', 'min': 0, 'max': inf, 'weight': 2}}

Parsing Stats file: Stats.txt

Number of events in Stats.txt: 5

Stats parsed successfully: {

'Logins': {**'mean'**: 4.0, **'std\_dev'**: 1.5},

'Time online': {'mean': 150.5, 'std\_dev': 25.0},

'Emails sent': {'mean': 10.0, 'std\_dev': 3.0},

'Emails opened': {'mean': 12.0, 'std\_dev': 4.5},

'Emails deleted': {'mean': 7.0, 'std\_dev': 2.25}}

Checking for inconsistencies...

Inconsistencies found: []

Parsed Events: {

'Logins': {'type': 'D', 'min': 0, 'max': inf, 'weight': 2},

'Time online': {'type': 'C', 'min': 0, 'max': 1440, 'weight': 3},

'Emails sent': {'type': 'D', 'min': 0, 'max': inf, 'weight': 1},

'Emails opened': {'type': 'D', 'min': 0, 'max': inf, 'weight': 1},

'Emails deleted': {'type': 'D', 'min': 0, 'max': inf, 'weight': 2}}

Parsed Stats: {

'Logins': {'mean': 4.0, 'std\_dev': 1.5},

'Time online': {'mean': 150.5, 'std\_dev': 25.0},

'Emails sent': {'mean': 10.0, 'std\_dev': 3.0},

'Emails opened': {'mean': 12.0, 'std\_dev': 4.5},

'Emails deleted': {'mean': 7.0, 'std\_dev': 2.25}}

Inconsistencies found: []

**1.2 Potential Inconsistencies**

The program detects potential inconsistencies between Events.txt and Stats.txt to ensure data integrity before proceeding with event generation and analysis.

* **Missing Event Entries**:
  + If an event is present in one file but missing in the other, it is flagged as an inconsistency.
* **Type Mismatches**:
  + **Discrete Events**: Discrete events are expected to have integer values for mean in Stats.txt. If a non-integer mean is detected, it is flagged.
  + **Continuous Events**: Continuous events should allow decimal values for mean, aligning with their non-discrete nature.
* **Invalid Ranges**:
  + The program ensures that each event’s min and max values are consistent. For instance, a min that exceeds max would be flagged.

**1.3 Example Output for Potential Inconsistencies**

1. **If the ‘Logins’ Line in the Stats.txt mean is a decimal value instead.**

For example :

Logins:4.5:1.5:

An error output will show this:

Inconsistencies found: ["Discrete event 'Logins' has non-integer mean in Stats.txt."]

1. **If there is a missing entry, where "Emails sent:10:3:" line is remove**

An error output will show this:

**Inconsistencies found: ["Event 'Emails sent' in Events.txt missing in Stats.txt."]**

1. **If there is a mismatch for a continuous event,**

For example :

Time online:150:25.00:

An error output will show this:

Inconsistencies found: ["Continuous event 'Time online' has integer mean in Stats.txt."]

**Section 2: Activity Engine and Logs**

**2.1 Event Generation Process**

**Statistical Consistency:**

Each event is generated to approximate its specified mean and standard deviation in Stats.txt. This ensures that the generated logs reflect realistic patterns that would be expected under normal conditions.

* **Discrete Events**: For events marked as discrete (D), values are generated as integers, representing counts or occurrences. These values are based on a normal (Gaussian) distribution centered around the mean, with variation defined by the standard deviation.
* **Continuous Events**: For continuous events (C), values are generated as decimal numbers, reflecting continuous measurement values (e.g., time spent online). Continuous values are rounded to two decimal places for readability.

**2.2 Log File Structure File Name:**

The log file is named event\_logs.json.

Format: The JSON format was chosen because it is both human-readable and easy to parse in Python, making it an effective choice for storing and retrieving structured event data.

The event\_logs.json Structure

[

    {

        "Logins": 4,

        "Time online": 146.86,

        "Emails sent": 5,

        "Emails opened": 6,

        "Emails deleted": 4

    },

    {

        "Logins": 5,

        "Time online": 132.21,

        "Emails sent": 9,

        "Emails opened": 12,

        "Emails deleted": 6

    },

**…**

**]**

**Analysis engine**

**Alert Engine**

[11/06/24]seed@VM:~/.../A3$ python3 ids.py Events.txt Stats.txt

Parsing Events file: Events.txt

Number of events in Events.txt: 5

Events parsed successfully: {'Logins': {'type': 'D', 'min': 0, 'max': inf, 'weight': 2}, 'Time online': {'type': 'C', 'min': 0, 'max': 1440, 'weight': 3}, 'Emails sent': {'type': 'D', 'min': 0, 'max': inf, 'weight': 1}, 'Emails opened': {'type': 'D', 'min': 0, 'max': inf, 'weight': 1}, 'Emails deleted': {'type': 'D', 'min': 0, 'max': inf, 'weight': 2}}

Parsing Stats file: Stats.txt

Number of events in Stats.txt: 5

Stats parsed successfully: {'Logins': {'mean': 4.0, 'std\_dev': 1.5}, 'Time online': {'mean': 150.5, 'std\_dev': 25.0}, 'Emails sent': {'mean': 10.0, 'std\_dev': 3.0}, 'Emails opened': {'mean': 12.0, 'std\_dev': 4.5}, 'Emails deleted': {'mean': 7.0, 'std\_dev': 2.25}}

Checking for inconsistencies...

Inconsistencies found: []

Parsed Events: {'Logins': {'type': 'D', 'min': 0, 'max': inf, 'weight': 2}, 'Time online': {'type': 'C', 'min': 0, 'max': 1440, 'weight': 3}, 'Emails sent': {'type': 'D', 'min': 0, 'max': inf, 'weight': 1}, 'Emails opened': {'type': 'D', 'min': 0, 'max': inf, 'weight': 1}, 'Emails deleted': {'type': 'D', 'min': 0, 'max': inf, 'weight': 2}}

Parsed Stats: {'Logins': {'mean': 4.0, 'std\_dev': 1.5}, 'Time online': {'mean': 150.5, 'std\_dev': 25.0}, 'Emails sent': {'mean': 10.0, 'std\_dev': 3.0}, 'Emails opened': {'mean': 12.0, 'std\_dev': 4.5}, 'Emails deleted': {'mean': 7.0, 'std\_dev': 2.25}}

Inconsistencies found: []

[11/06/24]seed@VM:~/.../A3$ python3 log\_generator.py

Traceback (most recent call last):

File "log\_generator.py", line 58, in <module>

log\_data = generate\_event\_data(events, stats, days)

NameError: name 'events' is not defined

[11/06/24]seed@VM:~/.../A3$ python3 log\_generator.py

Day 1 log: {'Logins': 4, 'Time online': 146.86, 'Emails sent': 5, 'Emails opened': 6, 'Emails deleted': 4}

Day 2 log: {'Logins': 5, 'Time online': 132.21, 'Emails sent': 9, 'Emails opened': 12, 'Emails deleted': 6}

Day 3 log: {'Logins': 1, 'Time online': 191.68, 'Emails sent': 10, 'Emails opened': 9, 'Emails deleted': 7}

Day 4 log: {'Logins': 5, 'Time online': 174.01, 'Emails sent': 12, 'Emails opened': 6, 'Emails deleted': 8}

Day 5 log: {'Logins': 4, 'Time online': 124.49, 'Emails sent': 8, 'Emails opened': 10, 'Emails deleted': 7}

Logs saved to event\_logs.json