You must write a program that calculates uptime for stations in a charging network. It will take in a formatted input file that indicates individual charger uptime status for a given time period and write output to standard-output (stdout).

Station Uptime is defined as the percentage of time that any charger at a station was available, out of the entire time period that any charger *at that station* was reporting in.

Code: chargeUptime.py

```
import sys
import datetime
def parse_input_file(file_path):
    try:
        with open(file_path, 'r') as file:
            lines = file.readlines()
            stations = {}
            reports = []
            section = None
            for line in lines:
                line = line.strip()
                if not line:
                    continue
                if line == '[Stations]':
                    section = 'stations'
                    continue
                elif line == '[Charger Availability Reports]':
                    section = 'reports'
                    continue
                if section == 'stations':
                    parts = line.split()
                    station_id = int(parts[0])
                    charger_ids = list(map(int, parts[1:]))
```

```
                stations[station_id] = charger_ids
            elif section == 'reports':
                parts = line.split()
                charger_id = int(parts[0])
                start_time = int(parts[1])
                end_time = int(parts[2])
                up = parts[3].lower() == 'true'
                reports.append((charger_id, start_time, end_time, up))
        return stations, reports
    except Exception as e:
        print("ERROR", file=sys.stderr)
        sys.exit(1)
def calculate_uptime(stations, reports):
    charger_uptime = {}
    charger_total_time = {}
    for charger_id, start_time, end_time, up in reports:
        duration = end_time - start_time
        if charger_id not in charger_uptime:
            charger_uptime[charger_id] = 0
            charger_total_time[charger_id] = 0

        charger_total_time[charger_id] += duration
        if up:
            charger_uptime[charger_id] += duration

    station_uptime = {}

    for station_id, charger_ids in stations.items():
        total_uptime = 0
```

```
        total_time = 0

        for charger_id in charger_ids:
            if charger_id in charger_uptime:
                total_uptime += charger_uptime[charger_id]
                total_time += charger_total_time[charger_id]

        if total_time > 0:
            uptime_percentage = (total_uptime * 100) // total_time
            station_uptime[station_id] = uptime_percentage
        else:
            station_uptime[station_id] = 0

    return station_uptime

def main():
    if len(sys.argv) != 2:
        print("ERROR", file=sys.stderr)
        sys.exit(1)

    input_file = sys.argv[1]
    stations, reports = parse_input_file(input_file)
    station_uptime = calculate_uptime(stations, reports)

    for station_id in sorted(station_uptime.keys()):
        print(f"{station_id} {station_uptime[station_id]}")

if __name__ == '__main__':
    main()
```

Output:

(.venv) PS C:\Users\Harshitha\PycharmProjects\chargeruptime> python chargeUptime.py C:\Users\Harshitha\PycharmProjects\chargeruptime\coding-challenge-charger-uptime-main\coding-challenge-charger-uptime-main\input_1.txt

0 100

1 0

2 100

(.venv) PS C:\Users\Harshitha\PycharmProjects\chargeruptime> python chargeUptime.py C:\Users\Harshitha\PycharmProjects\chargeruptime\coding-challenge-charger-uptime-main\coding-challenge-charger-uptime-main\input_2.txt

0 66

1 100