Bài tập lớn

Thiết kế chương trình phần mềm với các hàm và cấu trúc dữ liệu thích hợp để xử lý dữ liệu cảm biến.

I. Mô tả:

Sinh viên viết chương trình sử dụng ngôn ngữ C hoặc C++ với các hàm và cấu trúc dữ liệu thích hợp để mô phỏng dữ liệu cảm biến bụi PM2.5 đo nồng độ hạt bụi có kích thước < 2.5 microns trong không khí.

- Phạm vi đo: $0 \div 600\ \mu g/m^3$
- Độ phân giải: $0.1\ \mu g/m^3$

Các nhiệm vụ yêu cầu bao gồm:

1. Task 1:

Viết chương trình cho phép người dùng cung cấp số lượng cảm biến, tần số trích mẫu và khoảng thời gian đo bằng cách gõ lệnh trên command-line với cấu trúc câu lệnh như sau:

```
C:\\dust_sim -n [num_sensors] -st [sampling] -si [interval]
```

Trong đó:

- `dust_sim`: là tên file chương trình đã biên dịch
- `-n [num_sensors]` là cặp tham số đầu vào để cung cấp số lượng cảm biến, `[num_sensors]` cần được thay thế bởi một số cụ thể. Chương trình cần đưa ra thông báo lỗi nếu chỉ một trong 2 tham số này xuất hiện. Nếu cả 2 thông số này không xuất hiện trong câu lệnh command-line thì chương trình sẽ lấy số lượng cảm biến mặc định là 1.
- `-st [sampling]` là cặp tham số để cung cấp thời gian trích mẫu với `[sampling]` cần được thay thế bởi một số nguyên dương với đơn vị là giây, thời gian trích mẫu nhỏ nhất cho phép là 1 giây. Chương trình cần đưa ra thông báo lỗi nếu chỉ một trong 2 tham số này xuất hiện. Nếu cả 2 thông số này không xuất hiện trong câu lệnh command-line thì chương trình sẽ lấy tần số trích mẫu mặc định là 30 giây.
- `-si [interval]` là cặp tham số để cung cấp khoảng thời gian đo với `[interval]` cần được thay thế bởi một số nguyên dương đơn vị là giờ, khoảng thời gian mô phỏng nhỏ nhất là 1 giờ. Chương trình cần đưa ra thông báo lỗi nếu chỉ một trong 2 tham số này xuất hiện. Nếu cả 2 thông số này không xuất hiện trong câu lệnh command-line thì chương trình sẽ lấy tần số trích mẫu mặc định là 24 giờ.

Chương trình sẽ xuất ra một tập dữ liệu bao gồm định danh của cảm biến (sensor id), thời điểm đo (timestamp) mô phỏng và giá trị cảm biến mô phỏng (values), với thời điểm bắt đầu mô phỏng là thời điểm hiện tại lấy từ giờ của hệ thống (giờ trong máy tính) trừ đi khoảng khoảng thời gian mô phỏng.

- Số định danh (id) của cảm biến là các số từ 1 đến `num_sensors` với `num_sensors` là số lượng cảm biến mà người dùng cung cấp trong câu lệnh command-line, ví dụ `num_sensors = 10` thì chương trình sẽ tạo ra 10 cảm biến có id là 1, 2, 3, …, 10.
- Thời điểm đo (mô phỏng) có định dạng là YYYY:MM:DD hh:mm:ss, trong đó
  - YY – năm, MM – tháng , DD – ngày.
  - hh – giờ, mm – phút, ss – giây.

  Ví dụ: 2022:12:01 08:30:02
- Giá trị đo mô phỏng được tạo ra ngẫu nhiên là một số thực, với độ chính xác là 1 chữ số sau dấu phẩy.

Chú ý: thời gian mô phỏng không phải là thời gian thực mà là thời gian mô phỏng tính toán do đó, sinh viên không dùng hàm tạo trễ như sleep() hoặc các vòng lặp tạo trễ.

Dữ liệu mô phỏng xuất ra sẽ được lưu vào một file có tên là "dust_sensor.csv", nếu file đã tồn tại thì ghi đè lên file cũ. File này tuân theo định dạng CSV (comma-separated values), tức là các trường dữ

liệu cách nhau bởi dấu phẩy. Sinh viên có thể tham khảo thêm về định dạng csv ở link sau: https://www.ietf.org/rfc/rfc4180.txt

Ví dụ câu lệnh command-line: `C:\\dust_sim -n 3 -st 60 -si 10`

- Giả sử giờ hệ thống tại thời điểm chạy câu lệnh command-line là 2022:11:26 10:00:00, thời điểm bắt đầu mô phỏng sẽ là 2022:11:26 00:00:00. Thời gian mô phỏng bao gồm cả thời điểm bắt đầu và thời điểm chạy câu lệnh.
- Dữ liệu trong file "dust_sensor.csv" sẽ có dạng như sau:

```
id,time,values
1,2022:11:26 00:00:00, 50.1
2,2022:11:26 00:00:00,24.2
3,2022:11:26 00:00:00, 200.5
1,2022:11:26 00:01:00,100.2
2,2022:11:26 00:01:00,55.4
3,2022:11:26 00:01:00,160.9
…
1,2022:11:26 10:00:00,120.2
2,2022:11:26 10:00:00,90.4
3,2022:11:26 10:00:00,351.0
```

Trong đó dòng đầu tiên "id,time,values" là dòng tiêu đề của các trường dữ liệu.

2. Task 2:

Viết một chương trình xử lý dữ liệu trong một file csv có định dạng như ở task 1. Chương trình phải được chạy bằng câu lệnh command-line như dưới đây.

`C:\\dust_process [data_filename.csv]`

Trong đó:

- `dust_process`: là file chương trình đã biên dịch
- `[data_filename.csv]` là file csv chứa dữ liệu cảm biến bụi. Nếu người dùng không cung cấp tên file thì mà chỉ gõ `C:\\dust_process` thì chương trình sẽ sử dụng tên file mặc định là "dust_sensor.csv".

Ví dụ: `C:\\dust_process dust_sensor_ee3491.csv`

Chương trình phải có khả năng xử lý được ít nhất 10000 điểm dữ liệu, tức là file đầu vào `data_filename.csv` có thể chứa ít nhất 10000 dòng.

a. Task 2.1:

Giả sử môi trường cần đo có nồng độ bụi dao động trong khoảng từ $5 \div 550.5\ \mu g/m^3$, chương trình cần thực hiện việc kiểm tra dữ liệu hợp lệ trong file csv. Các giá trị đo nằm ngoài khoảng trên đều là giá trị dị biệt và cần loại bỏ. Các điểm dữ liệu dị biệt này cần phải được lưu trong một file csv có tên là "dust_outlier.csv", có định dạng như sau:

```
number of outliers: 3
id,time,values
1,2022:11:26 00:00:00,2.1
3,2022:11:26 19:03:00,-1
3,2022:11:26 21:06:00,500.2
```

Trong đó dòng đầu tiên được "number of outliers: X" với X là số lượng các điểm dữ liệu lọc ra từ file ban đầu, trong ví dụ trên thì X = 3.

Sử dụng các giá trị đo hợp lệ để thực hiện các phần tiếp theo từ task 2.2 đến hết.

b. Tasks 2.2:

Nồng độ bụi có thể được quy đổi về chỉ số chất lượng không khí (AQI – Air quality index) như sau:

Nồng độ c $[\mu g/m^3]$	$0 \leq c < 12$	$12 \leq c < 35.5$	$35.5 \leq c < 55.5$	$55.5 \leq c < 150.5$	$150.5 \leq c < 250.5$	$250.5 \leq c < 350.5$	$350.5 \leq c \leq 550.5$
AQI	0 ÷< 50	50 ÷ <100	100 ÷ <150	150 ÷ <200	200 ÷ <300	300 ÷ <400	400 ÷ 500
Cấp độ ô nhiễm	Good	Moderate	Slightly unhealthy	Unhealthy	Very unhealthy	Hazardous	Extremely hazardous

Tính nồng độ bụi trung bình theo từng giờ, ví dụ giá trị trung bình tại thời điểm 2022:11:26 02:00:00 là nồng độ bụi trung bình trong khoảng từ 2022:11:26 01:00:00 đến 2022:11:26 01:59:59 của cảm biến, xác định chỉ số AQI tương ứng với giá trị trung bình đó và cấp độ ô nhiễm. Lưu kết quả tính toán ra một file csv có tên là "dust_aqi.csv" có định dạng như sau:

```
id,time,values,aqi,pollution
1,2022:11:26 00:01:00, 50.1,137,Slightly unhealthy
2,2022:11:26 00:01:00,24.2,76,Moderate
3,2022:11:26 00:01:00, 200.5,250,Very unhealthy
1,2022:11:26 00:02:00,100.2,174,Unhealthy
2,2022:11:26 00:02:00,10.4,43,Good
3,2022:11:26 00:02:00,160.9,210,Very unhealthy
…
```

c. Tasks 2.3:

Xác định giá trị lớn nhất (max), nhỏ (mean), và giá trị nồng độ bụi trung bình (mean) đo được tại mỗi cảm biến và lưu kết quả vào file có tên là "dust_summary.csv" có định dạng như sau: Thời điểm tương ứng với các giá trị lớn nhất và nhỏ nhất trong file trên là thời điểm sớm nhất mà các giá trị này xuất hiện trong file đầu vào. Thời gian tương ứng với giá trị mean là khoảng thời gian mô phỏng.

```
id, parameters, time, values
1, max,2022:11:26 08:30:00,350.8
1, min,2022:11:26 09:31:03,5.6
1, mean,10:00:00 ,200.5
2, max,2022:11:26 08:35:00,300.8
2, min, 2022:11:26 09:32:03,15.6
2, mean,10:00:00, 110.5
3, max, 2022:11:26 09:05:02,120.6
3, min, 2022:11:26 09:21:03,20.8
3, mean,10:00:00,70.5
…
```

d. Task 2.4:

Dựa trên dữ liệu tính toán ở task 2.2, thống kê tổng số giờ ở mỗi cấp độ ô nhiễm đo được tại các node cảm biến và lưu vào một file có tên là "dust_statistics.csv".

```
id, pollution,duration
1,Good,2
1, Moderate,1
1, Slightly unhealthy,3
1,Unhealthy,0
1,Very unhealthy,2
1, Hazardous,2
1, Extremely hazardous,0
2,Good,1
2, Moderate,2
2, Slightly unhealthy,0
2,Unhealthy,4
2,Very unhealthy,1
2, Hazardous,0
2, Extremely hazardous,1
…
```

3. Task 3:

Giả sử ta cần gửi dữ liệu thu được ở task 2.2 qua một giao thức truyền thông bằng cách tạo lập một bản tin truyền thông là một chuỗi byte có cấu trúc như ở dưới đây

Start byte	Packet Length	ID	Time	PM2.5 concentration	AQI	Checksum	Stop byte
0x00 (1 byte)	1 byte	1 byte	4 bytes	4 bytes	2 byte	1 byte	0xFF (1 byte)

Trong đó ý nghĩa của các byte/nhóm byte như sau:

- Start byte (1 byte) là byte khởi đầu luôn có giá trị là 0x00.
- Stop byte (1 byte) là byte kết thúc luôn có giá trị là 0xFF.
- Packet length là độ dài của gói tin bao gồm cả start byte và stop byte.
- Id là số định danh của cảm biến luôn lớn hơn 0.
- Time là giá trị thời điểm đo theo định dạng thời gian trong hệ điều hành Unix được tính bằng giây.
- PM2.5 concentration là giá trị nồng độ bui, là một số thực 4 bytes (theo chuẩn IEEE 754).
- AQI là chỉ số chất lượng không khí, và là số nguyên 2 bytes
- Checksum là byte kiểm tra độ chính xác của dữ liệu trong gói tin được tính bằng mã bù 2 của các byte [packet length, id, time, PM2.5 concentration, AQI]

Khi người dùng viết câu lệnh command-line như dưới đây

```
C:\\ dust_convert [data_filename.csv] [hex_filename.dat]
```

thì chương trình sẽ:

- đọc từng dòng của file đầu vào,
- chuyển đổi dữ liệu sang dạng gói tin là một chuỗi byte như mô tả ở trên, mỗi byte cách nhau bởi dấu cách, các byte được ghi dưới dạng số hex.
- và ghi mỗi gói tin này trên một dòng trong file đầu ra tương ứng.
- nếu file `hex_filename.dat` đã tồn tại thì sẽ ghi đè lên file cũ.

Trong câu lệnh trên:

- `data_filename.csv` là file đầu vào đuôi CSV có định dạng như ở task 2.2
- `hex_filename.dat` là file đầu ra có dạng text, với phần mở rộng là dat

Ví dụ:

```
C:\\dust_convert dust_aqi.csv hex_packet_ee3492.dat
```

Dòng "1,2022:11:26 00:01:00, 50.1,137,Slightly unhealthy" trong file dust_aqi.csv sẽ được chuyển đổi thành:

00 0F 01 63 81 57 3C 42 48 66 66 00 89 9A FF

II. Yêu cầu kỹ thuật khác:

Chương trình chạy với command-line cần lưu lại các lỗi xảy ra vào 1 log file, với mỗi task ở mục I sẽ phải có 1 log file tương ứng đặt tên là task1.log, task2.log và task3.log. Mỗi một lỗi có thông báo lỗi được ghi trên một dòng của log file với định dạng như sau:

Error AB: DESCRIPTION

Trong đó

- AB là mã lỗi, là một số nguyên có 2 chữ số (nếu số < 10 thì sẽ ghi thêm số 0 phía trước, ví dụ 01, 02, …)
- DESCRIPTION là mô tả lỗi (khuyến khích ghi bằng tiếng Anh, không sử dụng tiếng việt có dấu).

1. Một số lỗi có thể gặp ở task 1:

- Sai câu lệnh command-line ví dụ thiếu 1 hoặc nhiều tham số. Thông báo lỗi có thể là: "Error 01: invalid command"
- Sai định dạng các thông số đầu vào, ví dụ số lượng cảm biến < 0. Thông báo lỗi có thể là "Error 02: invalid argument"
- File "dust_sensor.csv" đã tồn tại nhưng không cho phép ghi đè. Thông báo lỗi có thể là "Error 03: denied access dust_sensor.csv"

2. Một số lỗi có thể gặp ở task 2:

- File đầu vào `data_filename.csv` không tồn tại hoặc không cho phép truy cập. Thông báo lỗi: "Error 01: file not found or cannot be accessed"
- File đầu vào `data_filename.csv` có nội dung không phải theo định dạng csv như đã quy định. Thông báo lỗi "Error 02: invalid csv file"
- Lỗi dữ liệu trong file csv.
    - Tất cả các trường dữ liệu trên một dòng đều bị bỏ trống, ví dụ: ", , "
    - Id bị bỏ trống hoặc không hợp lệ, ví dụ "-1,2022:11:26 00:00:00, 50.1"
    - Thời gian bị bỏ trống hoặc không hợp lệ, ví dụ "1,2022:11:26 00:00:, 50.1"
    - Giá trị nồng độ bụi bị bỏ trống, ví dụ "1,2022:11:26 00:00:00, "

  Đối với lỗi dự liệu, thông báo lỗi phải chỉ rõ lỗi ở dòng nào với cấu trúc thông báo lỗi như sau: "Error 03: data is missing at line X" trong đó X là số dòng trong file đầu vào với dòng tiêu đề (tên các trường dữ liệu) "id,time,values" được coi là dòng số 0, dòng tiếp theo sau dòng tiêu đề là 1.

3. Một số lỗi có thể gặp ở task 3:

- File đầu vào `data_filename.csv` không tồn tại hoặc không cho phép truy cập. Thông báo lỗi: "Error 01: file not found or cannot be accessed"
- File đầu vào `data_filename.csv` có nội dung không phải theo định dạng csv như đã quy định. Thông báo lỗi "Error 02: invalid csv file"
- File đầu ra `hex_filename.dat` đã tồn tại và không cho phép ghi đè. Thông báo lỗi "Error 04: cannot override hex_filename.dat"
- Lỗi dữ liệu trong file csv.
    - Tất cả các trường dữ liệu trên một dòng đều bị bỏ trống, ví dụ: ", , , , "
    - Id bị bỏ trống hoặc không hợp lệ, ví dụ " ,2022:11:26 00:01:00, 50.1,137,Slightly unhealthy"
    - Thời gian bị bỏ trống hoặc không hợp lệ, ví dụ "1, , 50.1,137,Slightly unhealthy"
    - Giá trị nồng độ bụi và/hoặc AQI bị bỏ trống, ví dụ "1,2022:11:26 00:01:00, , ,Slightly unhealthy"

  Đối với lỗi dự liệu, thông báo lỗi phải chỉ rõ lỗi ở dòng nào với cấu trúc thông báo lỗi như sau:

"Error 03: data is missing at line X" trong đó X là số dòng trong file đầu vào với dòng tiêu đề (tên các trường dữ liệu) "id,time,values" được coi là dòng số 0, dòng tiếp theo sau dòng tiêu đề là 1.

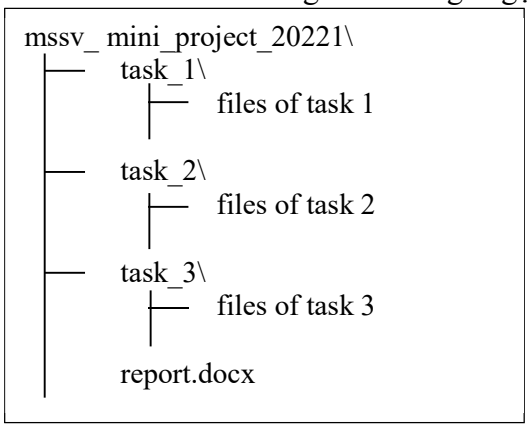
4. Sinh viên có thể tự đề xuất thêm các lỗi khác, tuy nhiên cần phải mô tả các lỗi khác đó trong báo cáo.

III. Thiết kế chương trình:

Sinh viên sử dụng các tiếp cận từ trên xuống (top-down approach) để thiết kế chương trình. Do đó, yêu cầu sinh viên phải vẽ sơ đồ top-down approach để minh họa cách phân hoạch hàm và quan hệ giữa các hàm trong chương trình trong báo cáo kèm theo mô tả/giải thích ngắn gọn.
Sinh viên cũng phải vẽ ít nhất 2 lưu đồ thuật toán:

- 1 lưu đồ thuật toán tổng quát cho toàn bộ chương trình
- Và 1 lưu đồ thuật toán cho 1 hàm quan trọng trong các hàm đã thiết kế (tùy chọn có thể vẽ nhiều hơn 1).

Sinh viên cũng phải vẽ sơ đồ cấu trúc thư mục và các file liên quan trong bài tập lớn theo định dạng như sau trong báo cáo cáo kèm theo mô tả/giải thích ngắn gọn.:



Trong đó thư mục gốc có tên là "mssv_ project_20221" và 3 thư mục con "task_1", "task_2" và "task 3" chứa các file liên quan. "report.docx" là file báo cáo và nằm trong thư mục gốc, "mssv" thay bằng mã số sinh viên.

IV. Coding styles (phong cách lập trình):

Coding style cần nhất quán trong toàn bộ chương trình và tuân theo quy định GNU mô tả trong link sau: https://www.gnu.org/prep/standards/html_node/Writing-C.html
Một cách ngắn gọn:

- Mã chương trình cần được trình bày gọn gàng, dễ theo dõi bằng cách lùi dòng, sử dụng dấu {}, ngắt dòng và cách dòng hợp lý.
- Cung cấp chú thích (comment) trong code rõ ràng, dễ hiểu để giải thích rõ hơn chương trình.
- Tên hàm và tên biến nên được đặt theo tiếng Anh, ngắn gọn và có tính tự mô tả.
- Tránh "hard-coding".

Lưu ý: Sinh viên cũng không được sử dụng các thư viện khác ngoài các thư viện chuẩn của C/C++.

V. Công cụ lập trình:

Editor: Visual studio code (https://code.visualstudio.com/download)
Compiler: gcc or g++ in MinGW-w64 (https://sourceforge.net/projects/mingw-w64/)
Sinh viên cũng cần mô tả rõ chương trình viết trong hệ điều hành nào (Windows, Linux, MacOS) trong báo cáo.

VI. Báo cáo và hướng dẫn nộp:

- Sinh viên làm bài tập theo cá nhân hoặc theo nhóm nếu phân theo nhóm.
- Toàn bộ bài tập lớn phải được tổ chức trong một thư mục như mô tả trong mục III.

- Sinh viên viết báo cáo là file **word** không quá 4 trang A4 (không bao gồm code chương trình và không nên đưa code vào báo cáo) sử dụng IEEE template (có đính kèm trong Team Assignment).
- Nội dung chính của báo cáo bao gồm:
    - Ý tưởng chính: mô tả ý tưởng thiết kế chương trình, bao gồm sơ đồ top-down approach, cấu trúc thư mục, các file code (nếu phân chia thành nhiều file code) và các thư viện được sử dụng.
    - Thiết kế chi tiết: mô tả thiết kế các hàm (nếu quá nhiều hàm thì cần mô tả chi tiết các hàm quan trọng nhất) bao gồm khuôn mẫu hàm (tên, kiểu trả về, danh sách tham biến) và mô tả đầu vào/ra. Chọn ít nhất 2 hàm quan trọng (trong đó 1 là luồng chương trình chính) và vẽ lưu đồ thuật toán.
    - Kết quả và đánh giá: mô tả kết quả chạy chương trình và đánh giá chất lượng chương trình
    - Kết luận: nêu văn tắt lại những vấn đề đã thực hiện được và chưa thực hiện được. Bảng đánh giá % đóng góp của mỗi thành viên trong bài tập lớp nếu làm theo nhóm.
    - Tài liệu tham khảo (nếu có).
- Sinh viên nén toàn bộ thư mục "mssv_mini_project_20221" thành file "mssv_mini_project_20221.zip" (chú ý là file .zip không sử dụng file .rar hay bất kỳ định dạng file nén nào khác).
    - Trong thư mục nộp bài chỉ giữ lại các file code, file chạy chương trình và file báo cáo bằng định dạng word (file docx).
    - Tất cả các file không liên quan đến bài tập lớn cần phải xóa trước khi nén và nộp.
    - mssv trong tên thư mục và tên file nén thay bằng mã số sinh viên

    Ví dụ: sinh viên Nguyễn Văn A có mã số sinh viên là 20221234 nộp bài "20221234_ mini_project_20221.zip"

    Nếu làm theo nhóm thì thay mssv bằng group_ID, ví dụ: "20221234_20222345_mini_project_20221.zip"
- Bài làm phải được nộp qua Team Assignment đúng hạn, không nộp bài qua email hay bất cứ kênh nào khác.

VII. Đánh giá:

- Bài tập lớn sẽ được chấm như sau:
    - Hoàn thành tất cả các tasks, chương trình chạy không có lỗi gì, xử lý tốt các trường hợp lỗi và có sáng tạo (4 điểm).
    - Thiết kế tốt và có tính sử dụng lại cao (2 điểm).
    - Phong cách lập trình tốt (good coding style) (2 điểm).
    - Báo cáo trình bày đúng template, bố cục rõ ràng, trình bày dễ hiểu, không có lỗi chính tả ngữ pháp (2 điểm).
- Sinh viên phải tự thực hiện bài tập lớn. Không copy bài của bạn khác và nên giữ bí mật bài làm của mình. **Nếu hai hay nhiều sinh viên có mã nguồn và/hoặc báo cáo giống nhau dù chỉ một phần thì bài làm của tất cả các sinh viên liên quan sẽ bị coi là pham quy và coi như không nộp bài không cần biết ai copy bài của ai.**
- Sinh viên chú ý nộp đúng hạn trên Team Assignment. Không được phép nộp muộn.
- Sinh viên (nhóm sinh viên) thực hiện tốt bài tập lớn sẽ được xem xét cộng điểm quá trình (1-2 điểm).
- **Sinh viên không nộp bài sẽ bị trừ 3 điểm quá trình.**

---------- Hết ----------