**Bài tập**: Viết chương trình đẩy dữ liệu từ cảm biến Senhah lên firebase, tối ưu thời gian cập nhật dữ liệu lên firebase tránh cập nhật dữ liệu không thay đổi gây dư thừa dữ liệu.

Các bước xây dựng:

**1. Tạo tài khoản và lấy token trong firebase**

|  |
| --- |
| config = {  "apiKey": "AIzaSyAGwKvxJeF4ezDYFYV\_mEeXnEWu8Zb\_dNE",  "authDomain": "duaniot12.firebaseapp.com",  "databaseURL": "https://duaniot12-default-rtdb.firebaseio.com",  "projectId": "duaniot12",  "storageBucket": "duaniot12.firebasestorage.app",  "messagingSenderId": "591100216879",  "appId": "1:591100216879:web:0e86adc53a185e15a928da",  "measurementId": "G-T5PC2KF844"  } |

**2. Khởi tạo Firebase và SenserHAT**

|  |
| --- |
| firebase = pyrebase.initialize\_app(config)  database = firebase.database()  sense = SenseHat() |

* **Khởi tạo Firebase**: Sử dụng thông tin cấu hình config để khởi tạo Firebase bằng thư viện pyrebase.
* **Khởi tạo SenseHAT**: Sử dụng thư viện sense\_emu để giả lập cảm biến SenseHAT. Đây là một cảm biến được sử dụng phổ biến trên Raspberry Pi để thu thập các dữ liệu về nhiệt độ, độ ẩm, và các cảm biến khác.

**3. Khai báo các biến toàn cục**

|  |
| --- |
| n = 5 # Kích thước lịch sử mảng  history = [0] \* n # Khởi tạo mảng lịch sử  previous\_T = 0 # Giá trị T trước đó  temperature\_change\_threshold = 1 # Ngưỡng thay đổi nhiệt độ (1 độ) |

**n**: Đây là số lượng dữ liệu được lưu trữ trong mảng history. Mảng này sẽ lưu lại các giá trị nhiệt độ đã đọc trong vài chu kỳ để tính toán giá trị trung bình.

**history**: Mảng để lưu trữ các giá trị nhiệt độ lịch sử, với kích thước n. Mảng này giúp tính toán giá trị nhiệt độ trung bình trong một khoảng thời gian.

**previous\_T**: Biến này lưu giá trị nhiệt độ trước đó để so sánh với nhiệt độ hiện tại và xác định xem có sự thay đổi lớn hơn 1 độ C hay không.

**temperature\_change\_threshold**: Đây là ngưỡng thay đổi nhiệt độ, được đặt là 1 độ C. Dữ liệu chỉ được gửi khi sự thay đổi nhiệt độ giữa lần đo trước và lần đo hiện tại vượt quá ngưỡng này.

**4. Hàm push\_optimized\_data()**

|  |
| --- |
| def push\_optimized\_data():  global history, previous\_T # Sử dụng biến toàn cục  while True:  try:  # Đọc nhiệt độ hiện tại từ SenseHAT  current\_temp = round(sense.get\_temperature(), 2) |

**Đọc nhiệt độ**: Sử dụng sense.get\_temperature() để lấy nhiệt độ hiện tại từ cảm biến SenseHAT và làm tròn giá trị đến 2 chữ số thập phân.

|  |
| --- |
| # Tính trung bình của lịch sử  mean\_temp = np.mean(history)  # Tính T\_cập\_nhật  T\_cap\_nhat = round((current\_temp + mean\_temp) / 2, 2) |

**Tính trung bình nhiệt độ**: np.mean(history) tính trung bình của các giá trị trong mảng history, giúp giảm thiểu sự sai lệch nếu cảm biến có độ lệch nhỏ trong mỗi lần đo.

**Tính T\_cập\_nhật**: T\_cap\_nhat là giá trị nhiệt độ cập nhật, được tính là trung bình của nhiệt độ hiện tại và trung bình của các giá trị trong lịch sử. Điều này giúp làm mượt dữ liệu.

|  |
| --- |
| # So sánh sự thay đổi nhiệt độ với ngưỡng  if abs(current\_temp - previous\_T) > temperature\_change\_threshold: |

**Kiểm tra sự thay đổi nhiệt độ**: So sánh sự thay đổi giữa nhiệt độ hiện tại và nhiệt độ trước đó (previous\_T). Nếu sự thay đổi lớn hơn ngưỡng 1 độ C, tiếp tục gửi dữ liệu lên Firebase.

|  |
| --- |
| # Gửi dữ liệu lên Firebase  sensor\_data = {  "temperature": T\_cap\_nhat,  "timestamp": time.strftime("%Y-%m-%d %H:%M:%S")  }  database.child("OptimizedSensorData").set(sensor\_data)  print("Đã gửi dữ liệu lên Firebase:", sensor\_data)  # Lưu vào SQL hoặc xử lý thêm tại đây nếu cần  previous\_T = T\_cap\_nhat # Cập nhật T |

**Gửi dữ liệu lên Firebase**: Nếu có sự thay đổi nhiệt độ lớn hơn 1 độ C, chương trình sẽ gửi dữ liệu lên Firebase, bao gồm nhiệt độ (T\_cap\_nhat) và thời gian (timestamp).

**Cập nhật previous\_T**: Sau khi gửi dữ liệu, previous\_T được cập nhật với giá trị của T\_cap\_nhat.

|  |
| --- |
| # Cập nhật mảng lịch sử  history.pop(0) # Xóa phần tử đầu tiên  history.append(current\_temp) # Thêm giá trị mới vào cuối  # In mảng lịch sử ra màn hình  print("Lịch sử nhiệt độ:", history)  # Tạm dừng 5 giây  time.sleep(5) |

**Cập nhật mảng lịch sử**: Mảng history được cập nhật sau mỗi chu kỳ. Phần tử đầu tiên trong mảng bị loại bỏ và giá trị mới được thêm vào cuối mảng.

**Tạm dừng 5 giây**: Chương trình tạm dừng trong 5 giây trước khi tiếp tục kiểm tra lại.

***Code chương trình***

|  |
| --- |
| import pyrebase  from sense\_emu import SenseHat  import time  import numpy as np  # Cấu hình Firebase  config = {  "apiKey": "AIzaSyAGwKvxJeF4ezDYFYV\_mEeXnEWu8Zb\_dNE",  "authDomain": "duaniot12.firebaseapp.com",  "databaseURL": "https://duaniot12-default-rtdb.firebaseio.com",  "projectId": "duaniot12",  "storageBucket": "duaniot12.firebasestorage.app",  "messagingSenderId": "591100216879",  "appId": "1:591100216879:web:0e86adc53a185e15a928da",  "measurementId": "G-T5PC2KF844"  }  # Khởi tạo Firebase và SenseHAT  firebase = pyrebase.initialize\_app(config)  database = firebase.database()  sense = SenseHat()  # Biến toàn cục  n = 5 # Kích thước lịch sử mảng  history = [0] \* n # Khởi tạo mảng lịch sử  previous\_T = 0 # Giá trị T trước đó  temperature\_change\_threshold = 1 # Ngưỡng thay đổi nhiệt độ (1 độ)  # Hàm đọc dữ liệu và tối ưu gửi  def push\_optimized\_data():  global history, previous\_T # Sử dụng biến toàn cục  while True:  try:  # Đọc nhiệt độ hiện tại từ SenseHAT  current\_temp = round(sense.get\_temperature(), 2)  # Tính trung bình của lịch sử  mean\_temp = np.mean(history)  # Tính T\_cập\_nhật  T\_cap\_nhat = round((current\_temp + mean\_temp) / 2, 2)  # So sánh sự thay đổi nhiệt độ với ngưỡng  if abs(current\_temp - previous\_T) > temperature\_change\_threshold:  # Gửi dữ liệu lên Firebase  sensor\_data = {  "temperature": T\_cap\_nhat,  "timestamp": time.strftime("%Y-%m-%d %H:%M:%S")  }  database.child("OptimizedSensorData").set(sensor\_data)  print("Đã gửi dữ liệu lên Firebase:", sensor\_data)  # Lưu vào SQL hoặc xử lý thêm tại đây nếu cần  previous\_T = T\_cap\_nhat # Cập nhật T  # Cập nhật mảng lịch sử  history.pop(0) # Xóa phần tử đầu tiên  history.append(current\_temp) # Thêm giá trị mới vào cuối  # In mảng lịch sử ra màn hình  print("Lịch sử nhiệt độ:", history)  # Tạm dừng 5 giây  time.sleep(5)  except Exception as e:  print("Lỗi xảy ra:", e)  # Chạy chương trình  if \_\_name\_\_ == "\_\_main\_\_":  print("Bắt đầu gửi dữ liệu tối ưu lên Firebase...")  try:  push\_optimized\_data()  except KeyboardInterrupt:  print("Đã dừng chương trình!") |

**Kết quả chương trình:**

A screenshot of a computer

Description automatically generated