For this assignment, Linked Lists are the preferred data structure. Hence, choose linked list (any type that you are comfortable with, singly linked list or circular linked lists or doubly linked lists) for any storage requirement you may have.

Physical Environment Sensor Network

A sensor network consists of spatially dispersed and dedicated sensors for monitoring and recording physical conditions of environment like temperature, sound, humidity, wind, and so on for multiple stations (you can consider different areas in city). The data from all sensors is collected at central location. Each sensor records the corresponding data at continuous time interval daily. Each sensor is represented by sensor ID (integer), sensor type, data it senses, time interval during which it senses the conditions continuously. Two sensors are said to be neighbours and can communicate if they are located at distance of less than 10Km. Solve the following question

1. Write a function **create_sensor_list(struct sensor_node* new_node)** which will formulate the above mentioned sensor network in software. Data fields for every sensor node in the linked list should have following attributes -
   a. **sensor_ID (integer)**
   b. **sensor_type (char)**
   c. **sensor_location(charater)(or sensor station)**
   d. **duration(time interval ex. 5 min- it means that a sensor senses the temperature , humidity etc. after every 5 min.)**
   e. **distance_between_stations (distance of current station from previous station and should be in km)**

   Note: First node in the list will have neighbour_distance 0

   Central repository is the location where data from all sensors is collected and it should include following things-

   a. **sensor_ID**
   b. **Date**
   c. **Time**
   d. **data(integer or float)**

   Write a function **central_repository (struct record* new_record)** which will create a database of information collected from all sensors.

2. Implement a function **Install_new_Sensor()** to add sensor in between stations as per following requirement
   A. If the distance between two stations is more than 10Km.
   B. If the type of sensor to be added is not present in between the stations.

3. Idle sensors are those which are not sending any information to central repository for more than 2months. Identify such idle sensors and remove them from database.

   (remove means permanently delete them from database)

4. **Retrieve_info**() functions retrieves the data for sensors specified by following conditions

   A. Depending on sensor type (retrieves till date data)

   B. Depending on specified date for specific sensor type

   C. Depending on specific time interval for specific sensor type

      a. for specified date (single day)

      b. for specified date range (multiple dates)

   Implement above functions for particular station as well as for all stations.

5. Write a function **find_communicating_neighbours** () which should find all the groups of neighbours. This function should check neighbourhood conditions mention above and should create the group of neighbours. Ex. If node A and B is 3Km apart. B and C is 2km apart. So the neighbours group will contain all nodes i.e. A, B and C.

6. Adapt the existing data structure for sensor type which records multiple quantities. Ex. **Air quality index (AQI)** sensor which records entities like PM10, PM 2.5, nitrogen dioxide, sulphur dioxide, carbon monoxide, ground level ozone etc. and tries to find out Air quality level and pollution level. AQI is measures based on the average quantity of a particular entity measured over a standard time interval. Standard time interval for measuring averages is different for different entities (24 hours for most of the entities, 8 hrs. For PM 2.5). There should be provision for storing standard time interval for each independent entity in existing data structure. Final AQI is the highest of the AQI values calculated separately for each entity. AQI value for finding health status is as follows

AQI	Status
1-50	Good
51-100	Satisfactory
101-200	Moderately polluted
201-300	Poor
301-400	May cause respiratory illness
401-500	Severe
501 onwards	Hazardous

   a. Write a function to report or display the month during which maximum

AQI is reported for all years for all stations.

b. Write a function to find out the date on which particular health status is recorded (input health status and station from user i.e. for which health status user wants to go)

c. Write a function to display the dates on which hazardous health status is recorded for all stations.