1. Write class Patient that will store information on the name, last name, age, height, and weigth of the person. Name and last name should be pointers to char strings. Height and weigth should be floating numbers. All aformentioned class members should be only accesible for the methods within the class.
NOTE: Please consider qualifiers for data types (e.g. short, long, unsigned) to reduce the number of bytes used to store information that has an upper limit e.g. will age be greater than 200 years? So maybe we do not need 4 bytes.

2. Write a constructor for class Patient that will take no arguments but will set all members fields to zeros or in terms of strings to an empty string " ".

3. Write a constructor that will take arguments (name, last name, age, weight, and height) and set values of corresponding class members. Name, last name and age are always required but when weight and height are not specified set them to zeros.

4. Write a copy constructor that will create a deep copy of all the data that were stored in one Person and write them to a new object of Patient.

5. Write a method that will calculate BMI for a Patient. Within a method check if all required information are stored in the Person object, if not return any number different then a BMI can have and write an info about that to CERR (stream of errors, from the same family as cout) or throw an exception.

6. Write all needed methods setting and getting values. Assume that name and last name should not be changed. In each method check if argument's value is correct, e.g. age cannot be lower than 0.

7. Write a function display with no arguments that will send to the output device all the information that a Patient object is storing in a specified format:
Name LastName (Age: ..., Heigth: ..., Weigth: ..., BMI: ...)

8.Write a custom implementation of the << operator for displaying information just as in the display function.

9. Write a destructor for class Patient. Destructor should also write to the output information: "Object of class Patient was destroyed"

10. Write class Clinic that will store name of the clinic, address and a list of pointers to registered patients (call it patients). Name should be a pointer to char string, but address might be

11. Write a constructor for class Clinic that will take as arguments name of the clinic and its address.

12. Write a copy constructor for class Clinic.

13. Write a method addPatient that will add a patient to a clinic (its pointer to the list of patients)

14. Write a method removePatient that will remove from the list a patient specified by its name and last name.

15. Write method checkBMI that will check BMI of a patient of a given name and last name. If specified patient is not registered in the clinic please write that on the output CERR. You can use calculateBMI from class patient but if a patient did not have weight and height specified ask a user to write those required information (CIN) and save them in the patient object.

16. Write method choosePatients that will dispaly name and last names of patients with BMI greater than value given by the argument.

17. Write a function displayPatients that will display all the information on

patients registered in the clinic using << operator implemented in the Patient class.

18. Write a destructor for a clinic in which you will iterate over the list and call destructors of patients patients registered in that clinic. At the very end destructor should send to the ouput information "Object of class Clinic was destroyed"

18. In the main function create at least 8 patients with all data set properly and 2 patient without height and weight specified. Use such weight and height values that all 8 patients will have different BMI with values from 16-35.

18. Create a Clinic and register all the patients in the clinic.

19. Display information on all Patients registered in the clinic

20. Display all the patients with BMI > 22

21. Remove from the clinic patients with BMI from 18.5 to 24.9.

22. Destroy objects that were previously removed from the clinic.

22. Display all the patients registered in the clinic.

23. Destroy the clinic with all patient objects registered in that clinic.