1. Define a class of banking Account with the following fields: an account number being a string, a balance being a real number and an array/list of operations of the account (empty by default) being real numbers: positive (incomes) or negative (outcomes). Implement the following public methods of the class:
    - the default constructor with the account number given as a parameter with default value being an empty string (the default balance of the account should be 0),
    - the copy-constructor and the assignment operator (=),
    - the destructor,
    - getNumber, setNumber and getBalance methods,
    - addOperation – adding a new operation for the account (the value of the operation should be stored in the list of operations as the last operation, and finally the balance of the account should be updated),
    - getIncomes – returning the sum of all the incomes of the account,
    - getOutcomes – returning the sum of all the outcomes of the account,
    - the indexing operator ([]) – returning the operation on the given position,
    - the shift operator (<<) – printing all the data of the account including the list of its subsequent operations.

    Write a program which tests all the class capabilities.
2. Define the Student class with the following fields: a forename, a surname (being strings), an index number being an integer number and an array/list of his marks (empty by default). Implement the following public methods of the class:
    - the default constructor with parameters for the first three fields of the student with some empty/zero default values,
    - the copy-constructor and the assignment operator (=),
    - the destructor,
    - set & get methods for the first three fields of the student (i.e., getSurname, setSurname, etc.),
    - addMark – adding a new mark for the student,
    - avgGrade – computing the average grade for the student by calculating the arithmetic mean of his marks,
    - the shift operator (<<) – printing all the data of the student.

    Define another class of the Group of students with a group number and an array/list of its students (it should be an array of pointers to the students, empty by default). Implement the following public methods of the class:
    - the default constructor with the number of the group given as a parameter with default value 0,
    - the copy-constructor and the assignment operator (=),
    - the destructor,
    - getNumber and setNumber methods,
    - addStudent – adding to the group a new student given as a parameter being a pointer to the existing student,
    - removeStudent – removing from the group the student with the index number given as a parameter,
    - count – returning the number of students in the group,
    - bestStudent – returning a pointer to such a student of the group with the highest average grade,
    - the shift operator (<<) – printing all the data of the group including the list of its students (optionally sorted by the student's surname).

    Test all the capabilities of the classes – create a group of students and let the user manage them by adding new students, adding their marks, removing students, counting them, obtaining the best student of the group and printing the group members.