Programming projects for *L4. Generic classes* (P175B123 EN)

Select the number of the project according the one given by the instructor in classrooms or online meetings.

!!! Contact your instructor for laboratory works in case you don't know or remember programming project number.

Tasks for *L4. Generic classes.* Requirements for particular programming project are provided in course Moodle file ***'Module description'*.**

Please analyze the mandatory requirements for each programming project (laboratory work):

- **!!! Check requirements in Moodle course file *'Module description'* regularly**
- **!!! Programming project can be defended only if the mandatory requirements for particular programing project are satisfied.**
- **!!! Report must be prepared according the given template.**

L4. Generic classes. *Mandatory prerequisites*:

1. All data collections (sets of values) are implemented as *generic* container class using doubly linked lists.
2. At least two *data classes:* must implement `IEquatable<T>`, `IComparable<T>` interfaces, auto-properties, constructor, overridden methods `Equals`, `ToString`, `CompareTo`.
3. Custom built (not system) *private sealed Node* generic class: stores data of single element, address to next element and address to previous element.
4. Container class must implement `IEnumerable<T>`, interface.
5. Usage of array (fixed length or dynamic e.g. `List`) or container class, implemented as an array, is not allowed.
6. Requirements for GUI:
   a) Creation of partial class *Form1*
   b) Program is managed by menu (component `MenuStrip`)
   c) Results are presented in screen using `RichTextBox`, `ListBox` components etc.
   d) Input data files are selected using `OpenFileDialog` control
   e) Result file is selected using `SaveFileDialog` control
7. Initial data and results are presented in a separate result text file **using table format**.

Initial data is given in two text files. Data is of different type.
Print method uses `IEnumerable` parameter for printing data *(-1 otherwise)*.
Separate form is created to display general information about the program. *(-0.5 otherwise)*.

L4-1. Publications. People order the press. Ordering takes place within a year. As a result, publications generate revenue (incomes). For each month find, which publication has the highest income (display: month number, name of publication). Calculate total incomes for publications (display: name of publication, total income). Make a list of publications with incomes below average (display: code, name, price for one month). Displayed list should be ordered by publication prices of one-month descending and publication name alphabetically. Create a separate list of subscribers by publication name (entered from keyboard) in defined month (entered from keyboard) (display: surname).

Data:

- Text file `U1a.txt` stores information about publications: code, name, price for one month.
- Text file `U1b.txt` stores information subscribers: surname, address, start of period (integer from 1..12), length of period, publication code, number of publications.

L4_2. Bonuses. Employees carried out work on 4 different topics. All topics received bonuses (money). Based on employee contributions, calculate the amount of bonus for each employee per topic and the total bonus amount (display: surname, name, bonus to first topic, bonus to second topic, bonus to third topic, bonus to fourth topic, total bonus). Don't give out more money than you have, and you must give out all the money. The employee's contribution shows what part of the bonus employee owns. The contribution can be expressed in any number, but the same metric must be used for all employers. Make a list of employees who earned less than average (display: surname, name, employees' contribution, amount of bonus for each topic, total amount of bonus). Printable list should be sorted by employee surname and name alphabetically. Create a list of earned bonus amount to each employee for specified topic (topic number is entered from keyboard) (display: surname, name, contribution to specified topic, bonus to specified topic, total bonus).

Data:

- Text file `U2a.txt` stores information about employees: id, surname, name, bank name, account.
- Text file `U2b.txt` stores information about employee contribution to work: first line holds bonus amounts (4 numbers). Following lines: employee id, employees' contribution to single topic, expressed as coefficient of efficiency. Information about individual employee is stored in one line.

L4_3. Modules. Students choose modules. Lecturers are responsible for organised modules. Individual lecturer may be responsible for several modules. Find which lecturer has the most selected modules and make a list of modules by such lecturer (display: module name, credits). List is sorted by number of credits in descending order and module name alphabetically. Find out whether students from all groups have selected modules for this lecturer i.e. one that has the most selected modules (display: `'Yes','No'`). Create a list of groups whose students did not select modules for this lecturer (display: group). Printed list should be ordered by group name in alphabetical order. Make a list of students who chose a particular module (module name is entered from keyboard) (display: student surname, student name, group) and sort it by surname and name alphabetically.

Data:

- Text file `U3a.txt` stores information about selected modules by students: module name, student surname, student name, and group.
- Text file `U3b.txt` stores information about modules: module name, surname of responsible lecturer, name of responsible lecturer, number of credits.

L4_4. Players. Basketball players are divided into 3 positions according to the position of the team: center (C), offender (O), defender (D). Basketball teams play matches with each other. Some teams win and others loose, there are no draws. Make lists of players for each position (display: team, name, surname, year of birth, height, number of matches played, points scored). Lists should be sorted by the number of points scored (descending) and the number of matches played (ascending). Find players of the team with most points (display: name, surname and player rank). Player rank is evaluated by calculating the ratio (points scored, number of matches played). Create a list of players for specified team `T` (display: name, surname, year of birth, playing position, number of matches played, points scored). Value of `T` is entered from keyboard.

Data:

- Text file `U4a.txt` stores information about basketball players: team, name, surname, year of birth, height (in meters), playing position, number of matches played, points scored.
- Text file `U4b.txt` stores information about teams: team name, number of matches played, number of matches won.

L4_5. Subscription. People order the press. Ordering takes place within a year. When ordering a press, an agent is assigned who will deliver the custom press home. Agents receive subscriber lists that are updated monthly. Display monthly subscriptions for each agent in each month (fields: agent code, subscription for month (12 values in total)). Create a list (display: address, surname, telephone, start of period, length of period, publication code, number of publications) of carried subscriptions for each agent in particular month `P` (month number is entered from keyboard). The printable list should be ordered by address of subscriber and surname alphabetically. Make a list of agents that carry more than the average value for a given month (display: agent code).

Data:

- Text file `U5a.txt` stores information about subscribers: address, surname, telephone, start of period (integer from 1..12), length of period (integer from 1..12), publication code, number of publications, agent code.
- Text file `U5b.txt` stores information about agents: agent code, surname, name, address, telephone.

L4_6. Factory. Every day the factory records information about parts produced by workers. The worker can produce different types of parts per day. Find the name of the most earned worker, count the number of days he worked, how many parts he produced and for what sum of money. Make a list of workers who produced only parts with same name (display: name and surname of worker, number of days worked, number of parts produced, sum of money). Create a list by filtering data from file `U6a.txt` according to criteria entered from keyboard (number of units produced `>` `S` and price `<` `K`). The printable list must be sorted by surnames and name alphabetically.

Data:

- Text file `U6a.txt` stores such information: year, month, day, name and surname of worker, part code, number of units produced.
- Text file `U6b.txt` stores such information: part code, part name, price for single unit.

L4_7. **Trigonometry**. File `U7a.txt` contains information about rectangles: name, coordinates (x, y) of upper left and lower right corners. File `U7b.txt` contains information about triangles: names, coordinates (x, y) of all 3 corners. Calculations:

a) Make a list of rectangles with exactly one triangle vertex inside the rectangle. Structure of such list is as follows: name of rectangle, coordinates of rectangle, name of triangle and coordinates of triangle corner. List must be sorted by name of triangle and rectangle.

b) Make a list of rectangles that have a triangle inside the rectangle. Structure of such list is as follows: name of rectangle, coordinates of rectangle, name of triangle and coordinates of all three corners. List must be sorted by name of triangle and rectangle.

c) Find one rectangle and triangle of greatest area. Compare their areas with each other.

During the defence, have a drawing corresponding to the information contained in the data files.

L4_8. Bus station. Buses carry passengers from the station. Some buses start the journey from the current station, whereas others are transit. Ticket price for transit buses are 10% lower. Find out if it is possible to travel from the current city to desired city `C` on the specified day of `D` in time interval range `[T1;T2]`. Values of `C`, `D`, `T1` and `T2` are entered from keyboard. If so, find the cheapest journey (display: city of departure, city of arrival, time of departure, time of arrival and price). Find the city to which most buses travel and create a separate list (display: city of departure, time of departure, city of arrival, time of arrival, day of week). Printable list must be sorted by city of departure alphabetically and time of departure in ascending order. Remove transit buses from created list.

Data:

- Text file `U8a.txt` stores information about routes for buses. Name of current station is specified in the first line of file. In the following lines: city of departure, time of departure, city of arrival, time of arrival, day of week.
- Text file `U8b.txt` stores information about prices for routes: city of arrival, price.

L4_9. Students. Students choose modules. Student can choose many modules. Lecturers are responsible for organised modules. Create a list of selected modules (display: module name, name, surname of responsible lecturer, number of credits). Create a separate list of modules that were not selected by the students (display: module name, name, surname of responsible lecturer, number of credits). Printable list should be ordered by module name and surname, name of the lecturer alphabetically. Find student that selected the most modules (display: student surname, student name). If multiple such students exist, find all. Create a list of modules, selected by particular student (values of student surname and name is entered from keyboard) (display module name).

Data:

- Text file `U9a.txt` stores information about selected modules by students: module name, student surname, student name, and group.
- Text file `U9b.txt` stores information about modules: module name, surname of responsible lecturer, name of responsible lecturer, number of credits.

L4_10. Scholarships. A fund of specified size is allocated for scholarships to students. Student receives a scholarship if the grade point average exceeds the specified average (given in first line of text file) and students has no debts (all grades > 4). A student receives a 10% higher scholarship if all his all grades are higher than 8. Such a student is called advanced. Assign the scholarships to students using presented fund (display: surname and name, grades, size of scholarship). Fund needs to be maximally distributed, but may not exceed the size of the fund. Sort created list by size of scholarship, surname and name. Remove students who do not get scholarships. Create a new list (display: surname and name, grades) of advanced students of the defined group `G`. Value of `G` is entered from keyboard.

Data:

- Text file `U10a.txt` stores information about personal data of students: id, surname and name, telephone number.
- Text file `U10b.txt` stores official information about students. Scholarship fund size and grade point average to receive the scholarship are specified in the first line of file. Following information is given in the following lines: id, group, number (count) of grade points, grades.

L4_11. Taxes. Residents pay utility taxes every month. Find which month and which tax cost the cheapest (display: name of month, tax code(-s)). Calculate the amount of money spent on taxes by all residents. Create a separate list of residents (display: surname and name, address) who paid less than average for utilities per year. Displayed list must be ordered by address, surname and name alphabetically. Remove from the list residents who did not pay for the specified tax `T` in the specified month `M` (`T`, `M` values are entered from keyboard).

Data:

- Text file `U11a.txt` stores information about utility services: tax code, tax name, the price of one unit of the service per month.
- Text file `U11b.txt` stores information about residents: name and surname of resident, address, month for which payment is made, tax code, the number of units consumed per month.

L4_12. Parts. Buyers order the devices needed for the production of robots in an online store. Find the most popular device(-s), how many such devices have been sold and for what amount of money. Create a list of buyers who have bought only one type of device (display: surname and name of buyer, number of devices purchased, and the amount of money paid for them). Displayed list must be ordered by the number of devices purchased and surname and name of buyer. Create a separate list of devices (display: device code, device name, device price) that at least *n* units have been sold and unit price of device does not exceed *k* euros. Values of *n* and *k* are entered from keyboard.

Data:

- Text file `U12a.txt` stores information about devices sold in an online shop: device code, device name, device price.
- Text file `U12b.txt` stores official information about buyers: surname and name of buyer, code of purchased device, number of purchased devices.

L4_13. Conversations. At the end of the month, the telephone service provider forms a report containing subscriber data: surname, first name, subscriber's phone number, called city, call start time (hour and minute), call duration in minutes. Calculate how much money each subscriber will have to pay for calls (display: surname, first name, phone number, amount paid). Displayed list must be ordered by subscriber's phone number in ascending order and surname, first name alphabetically. Call tariff (rate) is determined by the start time of the call. Create a separate list of subscribers (display: surname, first name, phone number, amount paid) who paid less money than average of all subscribers. Crate a list of subscribers who called the specified city `C` at the specified rate `R` (day or night) (display: surname, first name). Values for `C` and `R` are entered from keyboard.

Data:

- Text file `U13a.txt` stores information about subscribers: surname, first name, subscriber's phone number, called city, call start time (hour and minute), call duration in minutes.
- Text file `U13b.txt` stores information about prices of calls: city, price of connection (make a call), price of call for one minute at day (from 8 a.m. to 10 p.m.), price of call for one minute at night (from 10 p.m. to 8 a.m.).

L4_14. Hotels. Travelers choose hotels before summer vacations. Create a list of hotels chosen by travellers (display: name of hotel). Create a separate list of hotels not chosen by travellers (display: name of hotel). Create a list of travellers who plan to spend the most nights in hotels (display: surname and name of traveller). Displayed list must be ordered by surname and name of traveller. Make a list of travellers (fields: surname and name of traveller, amount of money) who paid a sum of money for hotels that does not exceed the amount indicated `M` (value entered from keyboard).

Data:

- Text file `U14a.txt` stores information about travellers: surname and name of traveller, name of chosen hotel, room type, planned number of nights.
- Text file `U14b.txt` stores information about hotels: name of hotel, room type, price for one night.

L4_15. Collectors. Collectors collect stamps. They exchange stamps with each other or buy stamps from each other at auctions. Find the most popular stamp (display: name of stamp). If multiple most popular stamps exist, find all. Find out if all collectors have at least one unit of this stamp (display: `'All','Not all'`). Create a list of collectors that does not have at least one unit of most popular stamp. (display name and surname of collector). Created list must be ordered by surname and name of collector. Create a separate list of collectors (display: surname and name of collector, name of stamp, number of stamps collected, sell price of stamp) who have stamps issued in year `Y` (value entered from keyboard) with the price of which does not exceed the average price offered at the auction. Created list must be ordered by name of stamp alphabetically and sell price of stamp in ascending order.

Data:

- Text file `U15a.txt` stores information about stamps: name of stamp, year of issue, average price offered at the auction.
- Text file `U15b.txt` stores information about collectors: name and surname of collector, name of stamp, number of stamps collected, sell price of stamp.

L4_16. Tasks. Instructors prepare laboratory work assignments for the spring semester. Assignments are given to students every month. Instructors receive list of students, which are updated monthly. Calculate the number of students supervised by each instructor in a given month `M` (entered from keyboard) (display: surname and name of instructor, number of supervised students). Create a separate list of instructors who supervised more students than average in given month `M` (display: surname and name of instructor, number of supervised students). Displayed list must be ordered by number of supervised students in ascending order and surname and name of instructor alphabetically. Create a separate list of assignments prepared by each instructor (display: surname and name of instructor, name of task(-s)). If there are instructors whose number of supervised students is less than specified (entered from keyboard), remove them from the list.

Data:

- Text file `U16a.txt` stores information about students: surname and name of student, group, month, task number.
- Text file `U16b.txt` stores official information about tasks: task number, name of task, surname and name of instructor, number of hours required to prepare the task.

L4_17. Project topics. Students choose topics for their project works. Instructors are responsible for project topics. Single instructor maybe responsible for several project topics. Create a list of instructors (display: surname and name of instructor). Created list must be sorted by surname and name alphabetically. Remove instructors from created list whose offered topics for projects were not selected by students. Calculate time spend for project topics of each instructor (display: name of instructor, surname of instructor, time spent). Create a separate list of project topics (display: name of project topic) of specified instructor (surname `S` and name `N` are entered from keyboard).

- Text file `U17a.txt` stores information about project topics selected by students: name of project topic, surname of student, name of student, group.
- Text file `U17b.txt` stores official information about project topics: name of project topic, surname of instructor, name of instructor, number of hours allocated to project topic.

L4_18. Instructors. Students choose modules. Instructors are responsible for modules. Single instructor maybe responsible for several modules. Create a list of instructors (display: surname of instructor, name of instructor). Created list must be sorted by surname and name in reverse alphabetical order. Remove instructors from created list whose modules were not selected by students. Find instructor(-s) whose modules were selected by the most students. Create a separate list of students (display: surname of student, name of student) of specified instructor (surname `S` and name `N` are entered from keyboard).

- Text file `U18a.txt` stores information about modules selected by students: name of module, surname of student, name of student, group.
- Text file `U18b.txt` stores official information about modules: name module, surname of instructor, name of instructor, credits.

L4_19. Products. The marketing departments of retail stores conduct product demand analysis. Find the most requested (most sought-after) product (-s) (display: name of product). Create a separate list of products which expire not earlier than after 30 days (display: name of product). Find which marketing store(-s) has the largest selection of products (display: name of store).

- Text file `U19a.txt` stores information about stores: name of store, name of product, product arrival date (year, month, day) in store, number of sold items (units), number of remaining items (units).
- Text file `U19b.txt` stores official information about products: name of product, expiration duration in days, unit price in euros (€).

L4_20. Football. Data about football players is provided. Create a list of football players who scored goals in range `[v-a, v+b]` where `v` is the average number of goals scored by one player per season (display: surname of player, name of player, name of team, scored goals). Find team(-s) players who scored the least goals (display: surname of player, name of player).

- Text file `U20a.txt` stores information about football players: surname of player, name of player, name of team.
- Text file `U20b.txt` stores official information about football player's game characteristics: surname, name, scored goals per season.