

Instruction manual of lite version of the program "Multilingual local instrumental system of public transport passenger delivery optimization, version 1.2"(LVP MLIS PTPDO 1.2)

Installing LVP MLIS PTPDO 1.2

1. Copy to a separate folder of your computer disk space a set of files of LVP MLIS PTPDO 1.2.
2. Check the composition of this set, which should include the following files:
 - boot file **Lite_MLIS_PTPDO.exe** of the program;
 - boot file **GLS.exe** of the utility program of version 1.0 under the name "**Генератор языковых оболочек (ГЯО 1.0)** (Generator of language shells (GLS 1.0))", which allows to create language shells for the LVP MLIS PTPDO 1.2 interface;
 - contents of the folder **Data** — data files with extension "dat" for several already solved demonstration tasks, next to which will be placed data files for tasks to be solved by the user;

Note

The installation set of LVP MLIS PTPDO 1.2 includes *ten* demonstration tasks, not available for removal, which are presented in two options: *Russian* ("DEMR01", "DEMR02", "DEMR03", "DEMR04") and *English* ("DEME01", "DEME02", "DEME03", "DEME04"). Input and output data of these tasks you can look through on the screen, as well as bring out to Excel-files or print

- contents of the folder **Dbrn** — files of databases of roads networks (DBRN1s), which are of the following four types: a) system-defined ordinary (sys1) and user-defined ordinary (use1) without information about houses; b) system-defined extended (sys2) and user-defined extended (use2) with information about houses. A file of DBRN1 of any type has the following name: **<three-digit digital database code><four-digit code of database type>_dbrn_<three-digit code of language of the current program shell .dat**;

Note

The installation set of LVP MLIS PTPDO 1.2 includes files of three test DBRN1s of various types: 1) a small database of the user-defined ordinary type (a file **001use1....dat**), 2) a large database of the user-defined ordinary type (a file **002use1....dat**) and 3) a small database of the system-defined extended type (a file **333sys2....dat**). All of them are presented in two options: *Russian* and *English*. Available for work is such group of these files which language

is the same as the program interface language you have selected. These databases were created in the program "Multilingual local instrumental system of transportation optimization (LVP MLIS TO)".

- contents of the folder **Dbtrn** — files of databases of routes networks (DBRN2s), to which are connected the required DBRN1s. The first are of three types: system, user and client. In LVP MLIS PTPDO 1.2, system DBRN2s can not be created or edited, user ones can, and client ones, formed from system ones, can only be created. A file of system/user/client DBRN2 has the following name: **sys/use/cli<three-digit digital database code>_dbtrn_<three-digit code of language of the current program shell>.dat;**

Note

The main purpose of client DBRN2s is to remotely solve tasks on public transport passenger delivery optimization using the multinational network instrumentation system (MNIS) PTPDO 1.2, when the initial DBRN2 of system type are the connected to it the DBRN1 of system-defined extended type (with information about houses) are located not at the network user, but on the server, which contains the remote optimization module of MNIS. Without the last one, such tasks could not be solved. In LVP MLIS PTPDO 1.2, client DBRN2s can also be used for solving tasks.

Note

The installation set of LVP MLIS PTPDO 1.2 includes files of four DBRN2s: **use001....dat**, **sys333....dat**, **cli333....dat** and **use002....dat**. The first three of them (small databases) are created on the basis of a Russian-language test map of networks of roads and routes (a file **КЦДМ_1.pdf**), and the last DBRN2 (large) is created without using a similar map, because there are no transport routes in it. All of them are presented in two options: *Russian* and *English*. Available for work is such group of these files which language is the same as the program interface language you have selected.

- contents of the folder **Documents of DBRN2 developer**:
 - files in the subfolder **rus**:
 - **КЦДМ_1_rus.pdf** — Russian-language map of networks of roads and routes, on the basis of which were created three of the four test DBRN2s of various typed (*see above*), attached to the installation set of the program;
 - files in the subfolder **eng**: absent;
- contents of the folder **Excel documents** — files of table format Excel (with extension "xls"), which, depending on their purpose, are distributed over the following two subfolders:
 - **Databases** — contains in nested folders **Rus** and **Eng** the files of databases of routes networks;
 - **Tasks** — contains in nested folders **Rus** and **Eng** the files of input and output data of any existing tasks, as well as files of output data of solved tasks;

Note

The installation set of LVP MLIS PTPDO 1.2 includes Excel-files with contents of three test databases of routes networks (folder **Database**), as well as with input data of four demonstration tasks (folder **Tasks**) in Russian and English options. These files you can not only view or edit in Microsoft Excel, but import to LVP MLIS when creating new DBRN2s and tasks.

- contents of the folder **Shells** — files of language shells for the current program interface;

Note

The installation set of LVP MLIS PTPDO 1.2 includes two files of shells for Russian and English languages: **Shell (PTPDO 1.2).rus** and **Shell (PTPDO 1.2).eng**.

- contents of the folder **Solving** — in this initially empty folder temporarily are placed data files for the tasks which are in the stage of solving;
 - contents of the folder **System**:
 - **info** — a text file containing three main parameters of the used computer: processor type, clock rate and the volume of operative memory;
 - **Languages.txt** — a text file containing a list of possible language shells for the interface of this program;
 - **Programs.txt** — a text file containing names of those programs (including this one), for which may be created language shells by using the utility program ГЯО 1.0 (GLS 1.0);
 - **protocol.prc** — file of a protocol of tasks, being solved by the user, which holds general information about all existing tasks and their current state;
 - contents of the folder **User documents**:
 - files of three Russian-language documents in the subfolder **rus**:
 - **Инструкция по эксплуатации ГЯО 1.0.pdf**;
 - **Инструкция по эксплуатации ОВП МЛИС ОДПГТ 1.2.pdf**;
 - **Технические характеристики МЛИС-МСИС ОДПГТ 1.2.pdf**;
 - files of three English-language documents in the subfolder **eng**:
 - **Instruction manual of GLS 1.0.pdf**;
 - **Instruction manual of LVP MLIS PTPDO 1.2.pdf** — file of this manual;
 - **Technical characteristics of MLIS-MNIS PTPDO 1.2.pdf**.
3. For ease of launching the program LVP MLIS PTPDO 1.2, create a shortcut for its file **Lite_MLIS_PTPDO.exe**, and place it on the desktop of your computer.

Run the program in operation

1. Click on the boot file **Lite_MLIS_PTPDO.exe** of the program or on its label. At that happens the following:
 - in the absence in the folder **Shells** of any language shells files appears on the screen a warning message that the work in the program LVP MLIS PTPDO 1.2 for this reason is impossible and it emergency closes;
 - in the presence in this folder of only one language shell file is launched the program, which interface is presented by the language of this shell;
 - in the presence there of two or more files of language shells happens transition to the next instruction item.
2. On the screen opens the window of the program LVP MLIS PTPDO 1.2 without any records of its interface, and in its center appears a small dialog under the name **Selection of program interface language**. From the drop-down list of this dialog select desired language of program interface and click the button **OK**. At that, the dialog closes and in the program window, which becomes available to work, appear necessary records in language that you just selected. In the same language will be output any text entries in all dialogs of the program, as well as all information provided in it.

Operations with tasks

Creating a new task with unique parameters

If input parameters of a new task of public transport passenger delivery optimization will differ significantly from similar parameters of any existing task, do the following:

1. Choose the command **Input of tasks**⇒**Input of a new task**, opening the dialog **Input of task data (step 1)**, wherein do the following:
 - enter a six-digit cipher of the new task, which may include Latin characters and digits (first field of the dialog);
 - enter a name of this task (second field);
 - choose the required database of routes network (drop-down list).

Note

The further order of setting parameters of the task depends on the type of the selected DBRN2. There are two possible options for subsequent setting of the parameters, each of which is realized when the corresponding condition is fulfilled (*see below*).

Condition of choosing DBRN2 of user type (a key phrase in the database name — "use")

2. Click in the current dialog the button **Forth**, passing to the dialog **Input of task data (step 2 of 4)**, in which specify the following parameters:

- allowable for use types of transport routes (five checkboxes);
- walking speed of a passenger (first field);
- optional parameter — allowable number of passenger boardings in different routes (second field);
- minimum allowable time of transition between two neighboring nodes of the routes network (third field).

Note

The last parameter is a system restriction that ensures the convergence of the used scheme of numerical optimization.

3. Click in the current dialog the button **Forth**, passing to the dialog **Input of task data (step 3 of 4)**, wherein specify addresses of the initial and final target points (TPs), namely:
 - type of street and its name (two drop-down lists, second and third fields of the table);
 - number of a house or sign of its absence (a dash) (fourth field of the table).
4. Click in the current dialog the button **Forth**. At that happens the following:
 - if for at least one target point is not specified its house number or this TP is located on a transit section with non-standard house numbering, then will open an additional dialog **Input of task data (step 3a of 4)**. Select in it those transit sections (TSs), on which must be located such TPs (sixth field of the table), then click the button **Forth**, passing to the dialog **Input of task data (step 4 of 4)** (*see below*);
 - in the above window, specify the second group of parameters for the target points, which include:
 - distance from the TP in yard to the exit on TS (sixth field of the table);
 - distance from the exit on a TS to its end (seventh field of the table);
5. Click in the current dialog the button **Forth**, passing to the dialog **Input of task data**. Look through in the above dialog the main input data of the task and if you are not satisfied by them, then using the button **Back** go to the desired previous dialog and make there necessary correction of the tasks input data. Otherwise, do one of two things:
 - at selected by default the switch **later** (it is at the bottom right) click the button **Ready**, causing the program to form two files in the folder **Data**: a common file of the task (**ptpdo_<six-digit task cipher>.dat**) and its input file (**ptpdo_<task cipher>_in.dat**);
 - select the switch **at once** (bottom left) and click the button **Ready**, causing this task to be solved at once without using stipulated for this aim the next commands: **Control**⇒**Work with a task protocol** and **Solving of tasks**⇒**Execution of optimization**.

Condition of choosing DBRN2 of system of client type (a key phrase in the database name — "sys" or "cli")

- Click in the current dialog the button **Forth**, passing to the dialog **Input of task data (step 2 of 3)**, in which specify the following parameters:
 - allowable for use types of transport routes (five checkboxes);
 - walking speed of a passenger (first field);
 - optional parameter — allowable number of passenger boardings in different routes (second field);
 - minimum allowable time of transition between two neighboring nodes of the routes network (third field).
- Click in the current dialog the button **Forth**, passing to the dialog **Input of task data (step 3 of 3)**, wherein specify addresses of the initial and final target points (TPs), namely: type of street, its name and house number (three drop-down lists and three last fields of the table).
- Click in the current dialog the button **Forth**, passing to the dialog **Input of task data**. Look through in the above dialog the main input data of the task and if you are not satisfied by them, then using the button **Back** go to the desired previous dialog and make there necessary correction of the tasks input data. Otherwise, do one of two things:
 - at selected by default the switch **later** (it is at the bottom right) click the button **Ready**, causing the program to form two files in the folder **Data**: a common file of the task (**ptpdo_<six-digit task cipher>.dat**) and its input file (**ptpdo_<task cipher>_in.dat**);
 - select the switch **at once** (bottom left) and click the button **Ready**, causing this task to be solved at once without using stipulated for this aim the next commands: **Control**⇒**Work with a task protocol** and **Solving of tasks**⇒**Execution of optimization**.

Creating a new task with repetitive parameters

If the input parameters of the new task will be almost the same as in some existing task, do the following:

- Choose the command **Input of tasks**⇒**Input of a new task by data import**, opening the dialog **Input of task data (step 1)**, wherein do the following:
 - enter a six-digit cipher of the new task (first field of the dialog);
 - enter a name of this task (second field);
 - select format of a being imported file which contains input data of another task: own format of the program (left switch **DAT** at the bottom) or standard format Excel (right switch **XLS** at the bottom);

Note

Mode of import into a new task of input data of another task, been stored in Excel-file, was introduced due to the fact that the program has a mode of export input data of existing tasks into files of Excel type (command **Export of the task data to Excel-table** of menu **Output of tasks**).

- when selecting the switch **DAT** do the following:
 - select in the second drop-down list the existing task, input data of which should be copied to the current task;
 - click the button **Import**, at that in the first drop-down list will display appears a name of the DBRN2 being used in this existing task;
 - when selecting the switch **XLS** do the following:
 - click the button **Import the Excel-file**;
 - in the opened dialog **Open** select on the disk the required file (with extension "xls"), and click on the button of the same name.
 - if you need, select a different DBRN2 than the one used in the imported existing task (first list).
2. Moving with the button **Forth** from the current dialog to a next one, look through in all dialogs the input data of a current task and make necessary changes in them (*see above Sec. "Creating a new task with unique parameters"*).
 3. Click the button **Ready** in the last dialog. At that, in the folder **Data** will be created two files of the task: general and input ones.

Correction of task input data

This operation can be performed in two ways:

- during the operation to create a new task (*see above*);
- by using the command **Input of tasks**⇒**Correction of input data of unsolved task**, that is available for use to such unsolved tasks, which input files have not been copied to the folder **Solving** for their solution (selection of a task to correct its input is made in the dialog **List of correctable tasks**, which opens on the screen when executing this command).

Uploading a file with input data of a new task

After creating a new task, you need to copy its input file (with input data of the task) to the folder **Solving** for its subsequent solution (*see below*). This operation is executed in the following order:

Note

After completing the operation in question concerning to some unsolved task you can not correct its input data.

1. Choose the command **Control⇒Work with a task protocol**, opening the dialog **Protocol of tasks**.
2. Select in the dialog upper list the required task, for which the operation in question was not fulfilled (this is indicated by the record "A stage of data input" in line "Task state" of the protocol table).
3. Click the button **Upload the file of task input data**, which in this case becomes unlocked. At that happens the following:
 - in the folder **Solving** appears a duplicate of the input file of current task;
 - in line "Task state" of the protocol table appears record "Waiting for solution";
 - the button **Upload the file of task input data** becomes locked.

Solving tasks

Operation of solving tasks, which input files are placed in the folder **Solving**, is performed in the following order:

1. Choose the command **Solving of tasks⇒Execution of optimization**, opening the dialog **Optimization module of MLIS PTPDO 1.2**.
2. Set the required mode of processing: *batch* (for solving all group of tasks) or *individual* (for solving only one task), and then click the button **Next**.
3. When working in batch mode, do the following:
 - if necessary, adjust specified parameters of optimization, and then click the button **Next**;
 - click the button **START** at the bottom, resulting in a process of successive solving the tasks of this group, when a name of this button changes to **WORK**;
 - after this button again be called **START**, which indicates the completion of solving the entire group of tasks, close the dialog **Optimization module of MLIS PTPDO 1.2** by clicking the button **EXIT** (bottom right).
4. When working in individual mode, do the following:
 - type in the dialog field a six-digit cipher of a solvable task and click the button **Next**;
 - if necessary, adjust the parameters of optimization scheme, and then click the button **Next**;
 - click the button **START** at the bottom, resulting in a process of solving selected task, when a name of this button changes to **WORK**;
 - after this button again be called **START**, which indicates the completion of solving the current task, close the dialog **Optimization module of MLIS PTPDO 1.2** (button **EXIT**).

Connecting a file with task solving results

After performing the operation of solving the tasks you should transfer files with the result of their solution (**ptpdo_<task cipher>_out.dat**) from the folder **Solving** to the folder **Data**. This operation performs in the following order:

Note

After performing this operation concerning some solved task it is impossible its repeated processing. That may be required, in particular, in case of absence of its successful solution due to wrong choice of values of the optimization schemes parameters.

1. Choose the command **Control⇒Work with a task protocol**, opening the dialog **Protocol of tasks**.
2. Select in the upper dialog list the required task, which input file was copied to the folder **Solving** (this is indicated by the record "Waiting for solution" in line "Task state" of the protocol table).
3. Click the button **Load the file of task solving results**, which in this case becomes unlocked. At that happens the following:
 - input file of the current task is removed from the folder **Solving**, and its output file is moved from this folder to the folder **Data**;
 - in line "Task state" of the protocol table appears record "Task is solved";
 - the button **Load the file of task solving results** becomes locked.

Working with a tasks protocol

A protocol of tasks contains general information about existing tasks of public transport passenger delivery optimization, being solved by LVP MLIS PTPDO 1.2 (these data are stored in the file **protocol.prc** placed in the folder **System**). On the screen displays in tabular form such part of the protocol, which refers to a user-selected task. It contains the following formation:

- cipher and name of this task;
- current state of the task, which can be one of three: "A stage of data input", "Waiting for solution" or "Task is solved";
- dates and times of occurrence for the four different event.

Transition to the mode of work with a tasks protocol is performed by the command **Control⇒Work with a task protocol**, which opens the dialog **Protocol of tasks**. In this dialog, the following operations can be performed:

- viewing general information about the existing tasks;
- upload the input file a new task, that contains its input data (*see above*);
- connecting the output file of the task containing the results of its solution (*see above*);
- removal of those tasks that are no longer necessary for the user (button **Delete the**

task).

Output of task input data

Input data for any existing task of public transport passenger delivery optimization may be brought out to the screen, Excel-file and print. This operation is performed in the following order:

1. Choose the command **Output of tasks**⇒**Viewing input data of any task**.
2. In the opened dialog **List of all tasks** select the desired task and click the button **OK**. At that, the current dialog is closed and opens the window **Viewing input data of the task "<task name>"** with tabular input data of the task chosen by you, which can be looked through.
3. To bring out these data to a new Excel document, choose the command **Export of the task data to Excel-table** of menu **Output of tasks**, to print them — the command **Print the task data** of the same menu.

Output of task solving results

Output data for any solved task public transport passenger delivery optimization may be brought out to screen, Excel-file and print. This operation is performed in the following order:

1. Choose the command **Output of tasks**⇒**Viewing output data of solved task**.
2. In the opened dialog **List of solved tasks** select the required task and click the button **OK**, closing this dialog. At that, opens the **Viewing the result of task solution "<task name>"** with tabular output data of the task, which can be viewed.
3. To bring out these data to a new Excel document, choose the command **Export of the task data to Excel-table** of menu **Output of tasks**, to print them — the command **Printing the task data** of the same menu.

Operations with databases of routes networks

Creating a new DBRN2 with unique parameters

If in no existing database of routes network was used that map of networks of roads and routes, on the basis of which a new DBRN2 will be created, perform the following actions:

1. Choose the command **Work with databases of routes networks**⇒**Creating a new DBRN2**, opening the dialog **Creating a database of routes network(step 1)**, wherein do the following:
 - enter a three-digit digital code of the new DBRN2 (first field of the dialog);
 - enter a name of this DBRN2 (second field).

- choose for connection the database of roads network (drop-down list).
2. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 2)**, in which specify the following parameters:
 - numbers of bus routes (first field of the dialog);
 - numbers of routes of fixed-route taxis (second field);
 - numbers of trolleybus routes (third field);
 - numbers of tram routes (fourth field);
 - numbers of metro lines (routes) (fifth field);
 - total number of ground transportation stops (GTSs) (sixth field);
 - total number of metro stations (MSs) (seventh field);
 - number of MS with transfers to other stations (eighth field);
 - total number of entrances-exits of metro (EEMs) (ninth field);
 - total number of entrances-exits of pedestrian crossings (EEMs) (tenth field);
 - mode of counter-clockwise numbering of exits from each pedestrian crossing or its absence (two switches on the right).

Note

The further order of setting DBRN2 parameters depends on the parameters specified in this dialog. There are four possible options for the subsequent steps to set parameters (*see below*): 1) in the absence of any transport; 2) in the presence of only ground transport; 3) in the presence of only underground transport (metro); 4) in the presence of ground and underground transports.

**Condition of absence of any transport
(zero value of fields 1–9 of the dialog)**

3. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 3 of 6)**, wherein specify the first group of parameters for location of transition objects:
 - number of EEMs on each transit section (third field of the table);
 - zero number of GTSs on TS (fourth field of the table);
 - zero number of EEMs on TS (fifth field of the table).
4. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 4 of 6)**, wherein specify the second group of parameters for location of transition objects:
 - identification number of a same-type transition object (fifth field of the table);
 - relative distance from the TS start to the transition object (sixth field of the table).
5. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database**

of routes network(step 5 of 6), wherein specify the number of pedestrian crossings from each EEPC (sixth field of the table).

6. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 6 of 6)**, wherein specify the second group of parameters for pedestrian crossings:
 - id. number of final EEPC (seventh field of the table);
 - average time of waiting for a start of pedestrian crossing (eighth field of the table);
 - length of pedestrian crossing (ninth field of the table).
7. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network**. Look through in it the main input data of the DBRN2 and if you are not satisfied by them, then using the button **Back** go to the desired previous dialog and make there necessary correction of the DBRN2 input data. Otherwise, press the button **Ready**, as a result of which the program will form in the folder **Dbtrn** the file of DBRN2 of the user type with the following name: **use<database code>_dbtrn_<code of the program language>.dat** (ordinary or extended).

Condition of availability only ground transport (zero value of fields 5, 7–9 of the dialog)

3. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 3 of 10)**, wherein specify the first group of parameters for location of transition objects:
 - number of EEPCs on each transit section (third field of the table);
 - number of GTSs on TS (fourth field of the table);
 - zero number of EEMs on TS (fifth field of the table).
4. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 4 of 10)**, wherein specify the second group of parameters for location of transition objects:
 - id. number of a same-type transition object (fifth field of the table);
 - relative distance from the TS start to the transition object (sixth field of the table).
5. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 5 of 10)**, wherein specify the number of pedestrian crossings from each EEPC (sixth field of the table).
6. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 6 of 10)**, wherein specify the second group of parameters for pedestrian crossings:
 - id. number of final EEPC (seventh field of the table);
 - average time of waiting for a start of pedestrian crossing (eighth field of the table);

- length of pedestrian crossing (ninth field of the table).
7. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 7 of 10)**, wherein specify the first group of parameters of ground transportation stops:
 - GTS name (fifth field of the table);
 - signs of using GTS by different types of transport vehicles (TVs) (next four fields of the table);
 - number of first following GTS for different transport routes (tenth field of the table).
 8. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 8 of 10)**, wherein specify the second group of parameters GTS:
 - id. number of next GTS (seventh field of the table);
 - durations of move to next GTSs by different types of TVs (next four fields of the table).
 9. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 9 of 10)**, wherein specify the first group of parameters of ground transport routes (TR):
 - TR number (third field of the table);
 - TR name (fourth field of the table);
 - a sign of circular TR (fifth field of the table);
 - period of arrival TV (sixth field of the table);
 - number of passable GTSs for direct half of TR (seventh field of the table);
 - number of passable GTSs for reverse half of TR (eighth field of the table).
 10. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 10 of 10)**, wherein specify the second group of parameters for ground TRs:
 - id. numbers of GTSs in order of their passing for direct half of TR (seventh field of the table);
 - the same for direct half of TR (eighth field of the table).
 11. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network**. Look through in it the main input data of the DBRN2 and if you are not satisfied by them, then using the button **Back** go to the desired previous dialog and make there necessary correction of the DBRN2 input data. Otherwise, press the button **Ready**, as a result of which the program will form in the folder **Dbtrn** the file of DBRN2 of the user type with the following name: **use<database code>_dbtrn_<code of the program language>.dat** (ordinary or extended).

Condition of availability only metro (zero value of fields 1–4, 6 of the dialog)

3. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 3 of 13)**, wherein specify the first group of parameters for location of transition objects:
 - number of EEPs on each transit section (third field of the table);
 - zero number of GTSS on TS (fourth field of the table);
 - number of EEMs on TS (fifth field of the table).
4. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 4 of 13)**, wherein specify the second group of parameters for location of transition objects:
 - id. number of a same-type transition object (fifth field of the table);
 - relative distance from the TS start to the transition object (sixth field of the table).
5. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 5 of 13)**, wherein specify the number of pedestrian crossings from each EEP (sixth field of the table).
6. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 6 of 13)**, wherein specify the second group of parameters for pedestrian crossings:
 - id. number final EEP (seventh field of the table);
 - average time of waiting for a start of pedestrian crossing (eighth field of the table);
 - length of pedestrian crossing (ninth field of the table).
7. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 7 of 13)**, wherein specify parameters of metro stations:
 - MS name (second field of the table);
 - duration of move to the next MS for the direct half of TR (third field of the table);
 - the same for the reverse half TR (fourth field of the table).
8. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 8 of 13)**, wherein specify the first group of parameters for transport routes (lines) of metro:
 - TR number (third field of the table);
 - TR name (fourth field of the table);
 - a sign of circular TR (fifth field of the table);
 - period of arrival TV (sixth field of the table);

- number of passable MSs (seventh field of the table);
9. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 9 of 13)**, wherein specify id. numbers of MSs in direct order of their passage (seventh field of the table).
 10. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 10 of 13)**, wherein specify the first group of parameters for transition of passengers inside metro:
 - id. number of a MS in transition start (second field of the table);
 - number of transition upon other MSs (third field of the table).
 11. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 11 of 13)**, wherein specify the second group of parameters for transition of passengers inside metro:
 - id. number of a MS in transition end (fifth field of the table);
 - length of pedestrian part of transition (sixth field of the table);
 - time of moving in the passage by escalators (seventh field of the table).
 12. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 12 of 13)**, wherein specify number of transitions to MSs from EEMs (fifth field of the table).
 13. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 13 of 13)**, wherein specify the second group of parameters for transitions between EEMs and MSs:
 - id. number MS (seventh field of the table);
 - length of pedestrian part of transition (eighth field of the table);
 - time of moving in the passage by escalators (ninth field of the table).
 14. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network**. Look through in it the main input data of the DBRN2 and if you are not satisfied by them, then using the button **Back** go to the desired previous dialog and make there necessary correction of the DBRN2 input data. Otherwise, press the button **Ready**, as a result of which the program will form in the folder **Dbtrn** the file of DBRN2 of the user type with the following name: **use<database code>_dbtrn_<code of the program language>.dat** (ordinary or extended).

Condition of availability ground and underground transports (positive values of fields 6, 7, 10 of the dialog)

3. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 3 of 17)**, wherein specify the first group of parameters for location of transition objects:
 - number of EEPs on each transit section (third field of the table);
 - number of GTSs on TS (fourth field of the table);

- number of EEMs on TS (fifth field of the table).
4. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 4 of 17)**, wherein specify the second group of parameters for location of transition objects:
 - id. number of a same-type transition object (fifth field of the table);
 - relative distance from the TS start to the transition object (sixth field of the table).
 5. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 5 of 17)**, wherein specify the number of pedestrian crossings from each EEPC (sixth field of the table).
 6. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 6 of 17)**, wherein specify the second group of parameters for pedestrian crossings:
 - id. number of final EEPC (seventh field of the table);
 - average time of waiting for a start of pedestrian crossing (eighth field of the table);
 - length of pedestrian crossing (ninth field of the table).
 7. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 7 of 17)**, wherein specify the first group of parameters of ground transportation stops:
 - GTS name (fifth field of the table);
 - signs of using GTS by different types of transport vehicles (TVs) (next four fields of the table);
 - number of first following GTS for different transport routes (tenth field of the table).
 8. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 8 of 17)**, wherein specify the second group of parameters GTS:
 - id. number of next GTS (seventh field of the table);
 - durations of move to next GTSS by different types of TVs (next four fields of the table).
 9. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 9 of 17)**, wherein specify the first group of parameters of ground transport routes (TR):
 - TR number (third field of the table);
 - TR name (fourth field of the table);
 - a sign of circular TR (fifth field of the table);
 - period of arrival TV (sixth field of the table);

- number of passable GTSs for direct half of TR (seventh field of the table);
 - number of passable GTSs for reverse half of TR (eighth field of the table).
10. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 10 of 17)**, wherein specify the second group of parameters for ground TRs:
- id. numbers of GTSs in order of their passing for direct half of TR (seventh field of the table);
 - the same for direct half of TR (eighth field of the table).
11. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 11 of 17)**, wherein specify parameters of metro stations:
- MS name MS (second field of the table);
 - duration of move to the next MS for the direct half of TR (third field of the table);
 - the same for the reverse half TR (fourth field of the table).
12. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 12 of 17)**, wherein specify the first group of parameters for transport routes (lines) of metro:
- TR number (third field of the table);
 - TR name (fourth field of the table);
 - a sign of circular TR (fifth field of the table);
 - period of arrival TV (sixth field of the table);
 - number of passable MSs (seventh field of the table);
13. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 13 of 17)**, wherein specify numbers of MSs in direct order of their passage (seventh field of the table).
14. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 14 of 17)**, wherein specify the first group of parameters for transition of passengers inside metro:
- id. number of a MS in transition start (second field of the table);
 - number of transition upon other MSs (third field of the table).
15. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 15 of 17)**, wherein specify the second group of parameters for transition of passengers inside metro:
- id. number of a MS in transition end (fifth field of the table);
 - length of pedestrian part of transition (sixth field of the table);
 - time of moving in the passage by escalators (seventh field of the table).

16. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 16 of 17)**, wherein specify number of transitions to MSs from EEMs (fifth field of the table).
17. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network(step 17 of 17)**, wherein specify the second group of parameters for transitions between EEMs and MSs:
 - id. number MS (seventh field of the table);
 - length of pedestrian part of transition (eighth field of the table);
 - time of moving in the passage by escalators (ninth field of the table).
18. Click in the current dialog the button **Forth**, passing to the dialog **Creating a database of routes network**. Look through in it the main input data of the DBRN2 and if you are not satisfied by them, then using the button **Back** go to the desired previous dialog and make there necessary correction of the DBRN2 input data. Otherwise, press the button **Ready**, as a result of which the program will form in the folder **Dbtrn** the file of DBRN2 of the user type with the following name: **use<database code>_dbtrn_<code of the program language>.dat** (ordinary or extended).

Creating a new DBRN2 with repetitive parameters

If a new database of routes network will be based on the same map of networks of roads and routes as some existing DBRN2, follow these steps:

1. Choose the command **Work with databases of routes networks⇒Creating a new DBRN2 by data import**, opening the dialog **Creating a database of routes network(step 1)**, wherein do the following:
 - enter a three-digit digital code of the new DBRN2 (first field of the dialog);
 - enter a name of this DBRN2 (second field).
 - select format of a being imported file of DBRN2: own format of the program (left switch **DAT**) or standard format Excel (right switch **XLS**);

Note

Mode of import into a new DBRN2 of contents of another DBRN2, been stored in Excel-file, was introduced due to the fact that the program has a mode of export contents of existing DBRN2 into files of Excel type (command **Export of DBRN2 to Excel-table** of menu **Work with databases of routes networks**).

- when selecting the switch **DAT** do the following:
 - select in the second drop-down list the existing DBRN2, contents of which should be copied to the current database;
 - click the button **Import**;
- when selecting the switch **XLS** do the following:

- click the button **Import the Excel-file**;
 - in the opened dialog **Open** select on the disk the required DBRN2 file (with extension 'xls'), and click the button of the same name.
 - if necessary, select another database of roads network for connection than the one connected to the imported DBRN2 (first list).
2. Moving with the button **Forth** from the current dialog to a next one, look through in all dialogs the data of a current DBRN2 and make necessary changes in them (*see above Sec. "Creating a new DBRN2 with unique parameters"*).
 3. Click the button **Ready** in the last dialog, whereby in the folder **Dbtrn** will be created a file of the new DBRN2 of the user type and the given language.

Correction of DBRN2 input data

This operation can be run in two ways:

- in the process of performing the operation on creating a new DBRN2 (*see above*);
- with the command **Work with databases of routes networks⇒DBRN2 correction** (choice of DBRN2 for its correction is made in the dialog **List of correctable databases of routes networks**, which opens on the screen when executing this command).

Note

The program LVP MLIS PTPDO 1.2 allows to create and correct DBRN2s of only user type. This is due to the fact that for system DBRN2s is provided centralized development with the purpose of their subsequent mass distribution, and client databases are derived from system databases.

Output of DBRN2 contents

Contents of any existing DBRN2 can be brought out on the screen and in the Excel-file. In the first case is used the command **Viewing of DBRN2** of menu **Work with databases of routes networks**, and in the second — the command **Export of DBRN to Excel-table** of the same menu.

Creating DBRN2 of a client type

This operation is performed using the command **Work with databases of routes networks⇒Creating a client DBRN2**. When it is executed, a dialog opens with a list of existing databases of routes networks having a system type, wherein you must select the database you need and click the button **OK**. As a result, from this database will be formed a file of the client DBRN2, which is unavailable for later editing.

Removing DBRN2

Deletion of any DBRN2 can be performed using the command **Deleting DBRN2** of menu **Work with databases of routes networks**.