Date: 21.10.2013 Name: Butum Claudiu Daniel

Problem statement:

Michael is the administrator of a bloc and wants to manage the monthly expenses for each apartment in the bloc. In order to complete this task, Michael needs an application to store, for a certain month, the expenses for each apartment. Each expense will be stored in the application through the following elements: amount, type of the expense (the administrator wants to group the expenses in several predefined categories: such as water, heating, illuminating, gas, others).

Bloc administrator - Feature list

F1. **Add** a new transaction into the list:
a) insert <amount>, <expense_type> at <apartment_id> – inserts at <apartment_id> an expense of <anmount> RON having the given <expense_type>

F2. **Modify** expenses from the list**:**
a) remove <apartment_id> – removes all the expenses at apartment <apartment_id>

b) remove from <apartment_id_low> to <apartment_id_high> – removes all the expenses from apartment <apartment_id_low> to apartment <apartment_id_high>.

c) remove <expense_type> -- removes all the expenses having the indicated <expnes_type>, from all the apartments.

d) replace <amount> , <expense_type> at <apartment_id> -- replaces the amount for the expense having the specified <expense_type> at apartment <apartment_id> with <amount> RON.

F3. **Write** the expenses having different properties.
a) list greater than <amount> -- writes all the apartments with an overall expense greater than <amount> RON

b) list less than <amount> before <apartment_id> -- writes all the apartments with an overall expense less than <amount> for apartments from 1 to <apartment_id>

c) list <expense_type> -- writes all the expenses having the specified <expense_type>

d) list sold|total <apartment_id> -- computes the sold (total amount) for apartment <apartment_id>
e) list – displays all the apartments

F4. **Obtain** different characteristics of expenses

a) sum <expense_type> -- writes the total amount for the expenses having the specified type **b)** max <apartment_id> -- writes the maximum expense per type for apartment 25 c) sort asc -- sorts the total expenses/apartment in an ascending order d) sort asc <expense_type> -- sorts the total expenses per <expense_type> in a ascending order e) sort desc -- sorts the total expenses/apartment in an descending order f) sort desc <expense_type> -- sorts the total expenses per <expense_type> in a descending order
F5. **Filter** **a)** filter <expense_type> -- retains only the expenses having the specified <expense_type>. **b)** filter <amount> -- retains only the overall expenses greater than <amount> RON
F6. **Undo** the last operation. **a)** undo – the last operation that has modified the list of expenses is cancelled.

Iteration plan

Iteration	**Planned features**
I1	F1, F2.a),d) F3.a),e)
I2	F2.b) c), F3.b) c) d), F4.a) b)
I3	F4.c) d) e) f), F5, F6

Iteration 1

Running scenario

User	**Program**	**Description**
	The help menu	The program shows the help menu with all the commands
insert 100, gas at 1	Success message	Inserts at apartmetn 1 an expense of gas = 100
list	The list of apartments.	Display the list of all apartments

remove 1		Remove all expenses from apartment 1
replace 1, heating with 100		Replaces the heating expense of apartment with 100 RON
list		
insert 200, others at 2		
list		
quit		Quits the program

Work items/tasks

T1	Filter the input from command line and get the specific command
T2	Implement user interface.
T3	Insert an apartment.
T4	Remove an apartment expenses.
T5	Replace an expense from the list
T6	Display all the apartments
T7	Display only apartments having a specific type

Iteration 2

Running scenario

User	**Program**	**Description**
list	the list of aparments	
help	help menu	Prints the help menu
list greater than 10	the list of apartments	The list of apartments with overall expense greater than 10
list gas	the list of apartments	List of apartments which has a gas type defined
list water	the list of apartments	List of apartments which has a water type defined

sum others	the total for 'others'	The total sum for expense 'others'
list total 2	the total	The total for apartment 2
max 2	the expense or expenses	The max expense for
list		
quit		Quit the program

Work items/tasks

T8	Remove expense from a range of apartments
T9	Remove all expenses with a given type
T10	Display filterd apartments
T11	Display the sum for an expense type
T12	Displays the maximum expense for an apartment

Iteration 3

Running scenario

User	**Program**	**Description**
list	the list of aparments	
help	help menu	Prints the help menu
filter gas	The list of apartments	The list of apartments which has the gas expense
insert 200, heating at 5		Insert at apartment 5 at heating an expense of 200
sort asc	The list sorted	The list sorted ascending by total expense
sort desc heating	The list sorted	The list sorted descending by heating
filter 200	Message	Reduce the list to only apartments with overall expense greater than 200

list	The list of apartments	
undo	Message	Undo the last operation
list	The list of apartments	
quit		Quit the program

Work items/tasks

T13	Implement sort ascending and descending by total expense
T14	Implement sort by type
T15	Filter the apartments
T16	Undo last operation