

Programming 2 – U1T2

Implement classes Tram which represents a tram described by selected data and class Depot which stores Trams.

For the classes declare and define methods such given main function will work.

(Do not modify main – except adding own tests if needed.)

Part 1 (2p)

Tram fields describe unique ID number of instance, current operating line number and home depot.

Implement constructor for Tram(char*) which:

- takes one parameter (char*) and set member depot_name[] to given parameter value,
- set line number to -1,
- give unique number based on total Trams counter, number cannot be changed after setting
- set status field to 'o'ff (character 'o')
- increment total trams counter

Class field status can be set to values (first character only stored) : 'r'eady, 'w'orking, 'f'ailure 'o'ff.

After constructing it should be set to off. Switching to states 'w'orking, 'f'ailure should increment total number of trams in use, being in 'o'ff and 'r'eady does not affect total number of trams in use.

Implement Methods:

- int getID() which return unique train number.
- char getState() - which returns state letter code
- int getInUse() – which inform about number of working trams;

Part 2 (2p)

Implement Tram class methods:

void Prepare(int) which perform:

- set line number
- set state to ready

void Run()

- increase number of trams in use
- if unique id is even set status to 'w'orking
- if unique id is odd set status to 'f'ailure

void returnDepot(char*) which set new depot Name;

Function void check(), which check state and in case of failure ask set new default Depot (= "Praga") for tram which state is : 'f'ailure

Part 2 (2p)

Implement class Depot and methods.

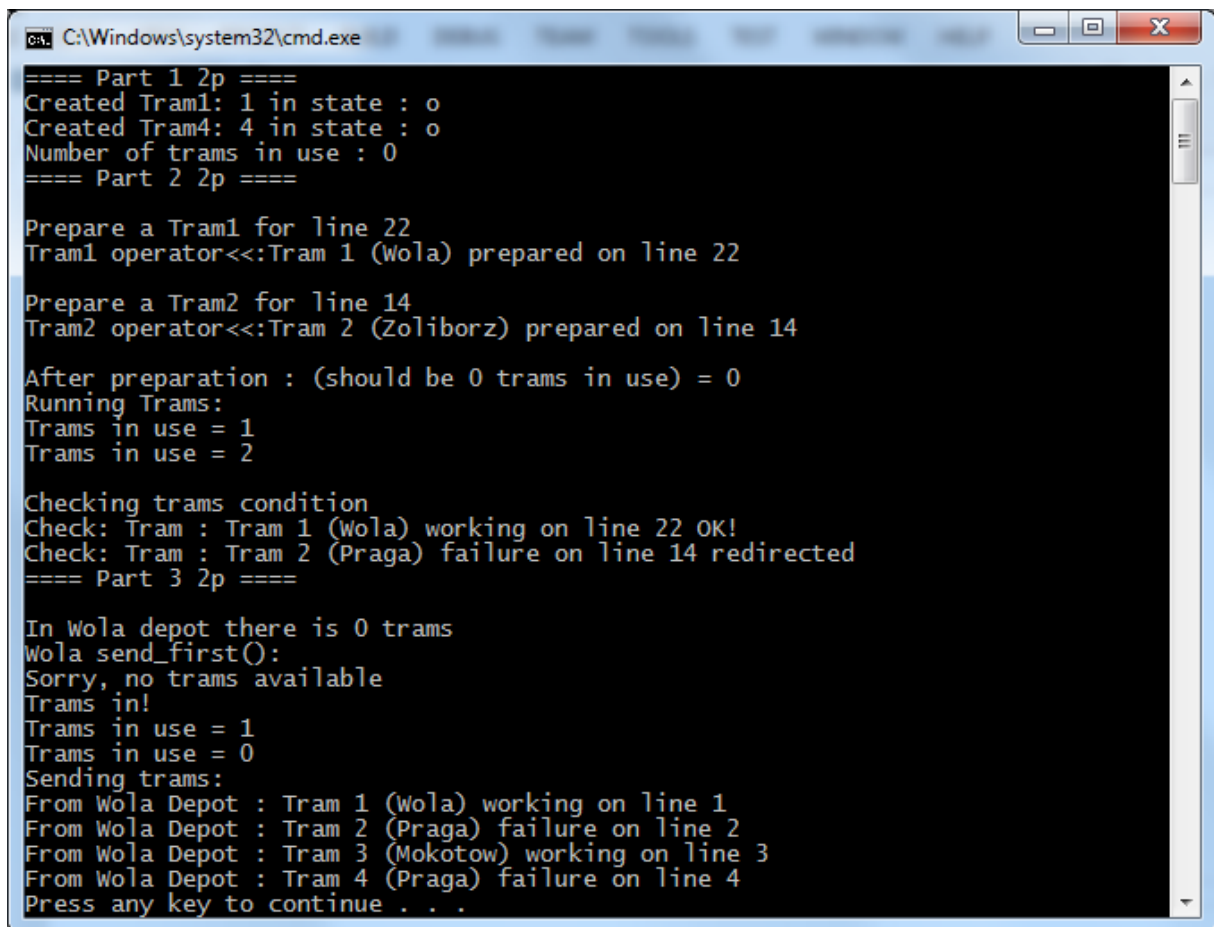
Class Depot's constructors sets name to depot instance and initialize „address slots” for trams;

Implement operator+= which add tram at the first free slot

Implement method send_first() which prepare and run tram from first not empty slot – lines numbers are incremented in subsequent method calls.

Implement operator<< which print information about Depot and trams which are currently in depot.

Example output:



```
C:\Windows\system32\cmd.exe

==== Part 1 2p ====
Created Tram1: 1 in state : o
Created Tram4: 4 in state : o
Number of trams in use : 0
==== Part 2 2p ====

Prepare a Tram1 for line 22
Tram1 operator<<:Tram 1 (Wola) prepared on line 22

Prepare a Tram2 for line 14
Tram2 operator<<:Tram 2 (Zoliborz) prepared on line 14

After preparation : (should be 0 trams in use) = 0
Running Trams:
Trams in use = 1
Trams in use = 2

Checking trams condition
Check: Tram : Tram 1 (Wola) working on line 22 OK!
Check: Tram : Tram 2 (Praga) failure on line 14 redirected
==== Part 3 2p ====

In Wola depot there is 0 trams
Wola send_first():
Sorry, no trams available
Trams in!
Trams in use = 1
Trams in use = 0
Sending trams:
From Wola Depot : Tram 1 (Wola) working on line 1
From Wola Depot : Tram 2 (Praga) failure on line 2
From Wola Depot : Tram 3 (Mokotow) working on line 3
From Wola Depot : Tram 4 (Praga) failure on line 4
Press any key to continue . . .
```