

Praktikum Objektorientierte Programmierung in C++ (WS 2021/2022)

[Dashboard](#) / [My courses](#) / [Wintersemester 2021/2022](#) / [Ingenieurwissenschaften](#) / [Informatik und Angewandte Kognitionswissenschaften](#)
/ [Praktikum OOP in C++ WS 2021/2022](#) / [Aufgabe 2/Task 2](#)
/ [A2 Teil 1: Hausaufgabe zur Vorbereitung auf die Präsenz-Gruppe/Part 1: Homework Task for Preparation of the Presence Group](#)

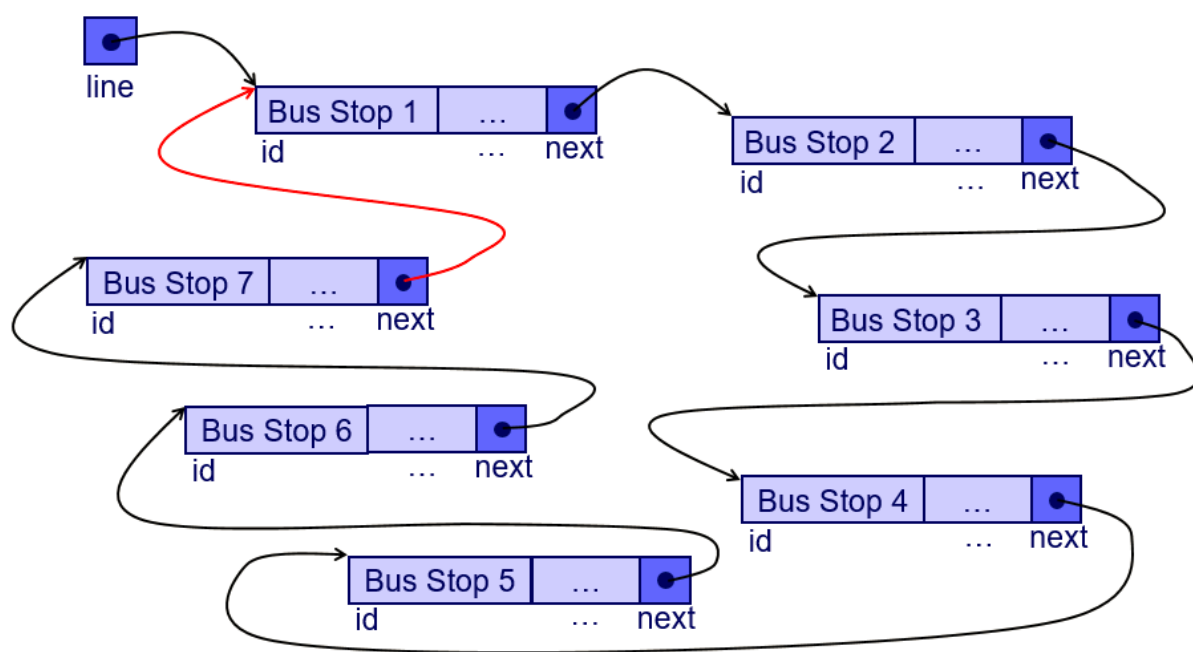
A2 Teil 1: Hausaufgabe zur Vorbereitung auf die Präsenz-Gruppe/Part 1: Homework Task for Preparation of the Presence Group

Ein Busunternehmen benötigt für die Planung eine (einfache) Simulation einer Bus-Ringlinie. Die Anzahl der Haltestellen der Ringlinie soll zu Beginn flexibel eingegeben, die Haltestellen der Reihe nach angefahren und von der letzten Haltestelle aus wieder die erste angefahren werden. In der Simulation sollen die Haltestellen als Listenstruktur erzeugt werden und das letzte Listenelement wieder auf das erste in der Liste zeigen (roter Pfeil statt Nullzeiger). Die Anzahl der an einer Haltestelle aus- und einsteigenden Fahrgäste soll über Zufallszahlen simuliert werden. Die Anzahl der Fahrgäste im Bus soll im Programm mitgezählt und gespeichert werden sowie für jede Haltestelle die Anzahl erfolgter Busstops und die Summe der Fahrgäste, die ein- und die ausgestiegen sind.

Achtung: Lesen Sie die Aufgabe zuerst sorgfältig und vollständig durch. Achten Sie vor allem darauf, welche Elemente die Datenstruktur (die zirkulär verkettete Liste) bilden und welche nicht./

A bus company needs a (simple) simulation of a bus circular line for planning. The number of stops of the circular line is to be entered flexibly at the beginning, the stops are to be approached in sequence and the first stop is to be approached again from the last stop. In the simulation, the stops are to be generated as a list structure and the last list element is to point to the first in the list again (red arrow instead of null pointer). The number of passengers getting off and on at a stop is to be simulated using random numbers. The number of passengers on the bus shall be counted and stored as well as for each stop the number of simulated bus stops and the total number of passengers who got on and off the bus.

Attention: Make sure to first read the task carefully and completely. Pay attention especially to which elements form the data structure (the circular linked list) and which of them do not.



Programmieren Sie im einzelnen:/Program in detail:

- Definieren Sie eine C++-Struktur (also ohne **typedef**) mit Namen **Bus** mit zwei Komponenten zur Speicherung der jeweils aktuellen Anzahl an Fahrgästen und der Maximalanzahl von Fahrgästen im Verlauf der kompletten Simulation.
Define a C++ structure (i.e. without **typedef**) called **Bus** with two components to store the current number of passengers and the maximum number of passengers during the complete simulation.
- Schreiben Sie eine kleine Funktion namens **init** ohne Rückgabe, die eine Referenz auf eine solche Strukturvariable **Bus** als einzigen Parameter hat. Setzen Sie Im Rumpf beide Komponenten der übergebenen Referenzvariable auf den Bus auf den Wert 0./
Write a function called **init** without return that has a reference to such a structure variable **Bus** as its only parameter. Assign in the body value 0 to both components of the passed bus reference variable.
- Schreiben Sie eine Funktion namens **output** ohne Rückgabe, die eine Referenz auf eine solche Strukturvariable **Bus** als einzigen Parameter hat. Geben Sie Im Rumpf die beiden Werte der Komponenten der Referenz auf den Bus wie im Beispiel unten (Menüpunkt 1) aus./
Write a function called **output** without return that has a reference to such a structure variable **Bus** as its only parameter. Output in the body both values of the components of the passed bus reference like in the example below (menue entry 1).

4. Definieren Sie eine weitere C++-Struktur (also ohne **typedef**) mit Namen **Stop** (Haltestelle) mit folgenden Komponenten:

- eine C++-Zeichenkette für die Bezeichnung der Haltestelle,
- drei ganze Zahlen für die Gesamtanzahl von Stops eines Busses an dieser Haltestelle sowie der Anzahl der insgesamt ein- und insgesamt ausgestiegenen Fahrgäste,
- ein Zeiger auf die nächste anzufahrende Haltestelle für den Bus./

Define a C++ structure (i.e. without **typedef**) called (bus) **stop** with the following components:

- a C++ string for the name of the stop,
- three integers: first integer is for the total number of times that any bus has stopped at the this station/bus stop. and the second and the third integers are for the total number of passengers who got on and got off the bus at the stop (respectively),
- a pointer to the next bus stop to be approached for the bus.

5. Schreiben Sie eine Funktion namens **generate_stop** mit einem Zeiger auf eine solche Haltestelle als Rückgabe, die eine C++-Zeichenkette als einzigen Parameter hat. Erzeugen Sie im Rumpf eine neue Haltestellen-Variable auf dem Heap, setzen Sie die Bezeichnung der Haltestelle auf den übergebenen Parameter, die drei Zahlenwerte auf 0 und den Zeiger auf die nächste Haltestelle auf einen C++-Nullzeiger./

Write a function called **generate_stop** with a pointer to such a stop as return, which has a C++ string as its only parameter. In the body, create a new stop variable on the heap, set the name of the stop to the passed parameter, the three numerical values to 0 and the pointer to the next stop to a C++ null pointer.

6. Schreiben Sie eine Funktion namens **generate_circular_line** (erzeuge Ringlinie) mit einem Zeiger auf eine Haltestelle als Rückgabe, die eine (ganzzahlige) Anzahl n von zu erzeugenden Haltestellen der Ringlinie als einzigen Parameter hat. Erzeugen Sie in einer Schleife im Rumpf n neue Haltestellen-Variable über die obige Funktion **generate_stop** und verketteten diese in einer Ring-Liste auf dem Heap wie in der Grafik oben gezeigt. Geben Sie einen Zeiger auf die erste Haltestelle als Funktionswert zurück.

Hinweis: für die Bezeichnung der k-ten Haltestelle können Sie jeweils die konstante Zeichenkette "**Bus Stop**" konkatenieren mit einer Zahl am Ende über einen Funktionsaufruf **to_string(k)**, der die Zahl als Zeichenkette zurück gibt (diese vordefinierte Funktion müssen Sie je nach IDE/Compiler evtl. über die Bibliothek **#include <string>** am Anfang der Datei mit einbinden)./

Write a function called **generate_circular_line** with a pointer to a stop as return, which has an (integer) number n as only parameter. This parameter represents how many stops should exist in the circular line. In a loop in the body, create n new stop variables calling above function **generate_stop** and concatenate them as a circular-list on the heap as shown above in the graphic. Return a pointer to the first stop as function value.

Hint: for the name of the k-th stop, you can have a constant "**Bus Stop**" string and concatenate numbers to the end of the string by a function call **to_string(k)**, which returns the number as a string (depending on the IDE/compiler, you may have to include this predefined function via the library **#include <string>** at the beginning of your file).

7. Schreiben Sie eine Funktion namens **output** ohne Rückgabe, die einen Zeiger auf eine solche Strukturvariable **stop** als einzigen Parameter hat. Geben Sie im Rumpf die Werte der Komponenten der Bushaltestelle wie im Beispiel unten (Menüpunkt 2) aus./

Write a function called **output** without return that has a reference to such a structure variable **stop** as its only parameter. Output in the body the values of the components of the bus stop like in the example below (menu entry 2).

8. Schreiben Sie eine überladene Funktion namens **output_whole_line** ohne Rückgabe, die einen Zeiger auf eine solche Strukturvariable **stop** als einzigen Parameter hat. Geben Sie im Rumpf alle Bushaltestellen wie im Beispiel unten (Menüpunkt 2) aus. Rufen Sie dabei in einer Schleife obige Funktion **output** für jede Bushaltestelle in der Ringlinie auf./

Write an overloaded function called **output_whole_line** without return that has a reference to such a structure variable **stop** as its only parameter. Output in the body all bus stop like in the example below (menu entry 2). Call above function **output** in a loop for each bus stop in the circular line.

9. Schreiben Sie eine Funktion namens **simulate_get_on_off** ohne Rückgabe, die als vier Parameter eine Referenz auf eine Strukturvariable **Bus**, einen Zeiger auf eine Strukturvariable **stop**, die Anzahl aussteigender und die Anzahl einsteigender Fahrgäste in den Bus an dieser Haltestelle hat. Überprüfen Sie im Rumpf zuerst, dass die Anzahl aussteigender Fahrgäste, die ja in der Simulation als Zufallszahl erzeugt wird, nicht größer ist als die Anzahl aktueller Fahrgäste im Bus. Ansonsten setzen Sie die Anzahl aussteigender Fahrgäste genau auf die Anzahl der Fahrgäste im Bus, um keine negative Anzahl an Fahrgästen im Bus zu bekommen. Geben Sie dann die Zeichenkette **simulate** aus, dahinter die Bezeichnung der Haltestelle und die Anzahlen aus- und einsteigender Fahrgäste (siehe Ausgaben zu Menüpunkt 3). Aktualisieren Sie danach die Anzahl der Fahrgäste im Bus, die Summenwerte der Stops (siehe Aufgabe 4, erste ganzzahlige Komponente), aus- und eingestiegenen Fahrgäste an dieser Haltestelle und ggf. auch die maximale Anzahl an Fahrgästen, die während der Simulation im Bus gegeben ist./

Write a function called **simulate_get_on_off** without return that has as four parameters a reference to a structure variable **Bus**, a pointer to a structure variable **stop**, the number of passengers getting off and the number of passengers getting on the bus at this stop. First check in the body that the number of passengers getting off the bus, which is generated as a random number in the simulation, is not greater than the number of current passengers on the bus. Otherwise set this number of passengers getting off the bus exactly to the number of passengers on the bus in order not to get a negative number of passengers on the bus. Then output after string **simulate** the name of the stop and the numbers of passengers getting off and on (see outputs for menu item 3). Afterwards update the number of passengers in the bus, increase the total number of times a bus stopped at the station (see Task 4, first integer component), as well as the number of people who got off and on at this stop and, if necessary, update the maximum number of passengers of the bus during the simulation (see menu entry 3).

10. Schreiben Sie eine **main**-Funktion, in deren Rumpf u.a. eine Strukturvariable für einen Bus und zwei Zeigervariable auf Haltestellen (**stop**) definiert werden, eine für die erste der Bus-Ringlinie und die zweite für die jeweils aktuelle Haltestelle, an der der Bus simuliert hält.

Setzen Sie

- die Werte der Komponenten von dem Bus über einen Aufruf der Funktion **init** und
- nach einer Eingabeaufforderung für die Eingabe der Anzahl der Haltestellen beide Zeigervariable auf den Rückgabe-Zeiger Ihrer Funktion **generate_circular_line**, also die erste Haltestelle in der erzeugten Bus-Ringlinie.

Über eine Menüauswahl soll danach die Fahrt eines Busses von Haltestelle zu Haltestelle simuliert und Ausgaben gemacht werden wie im Beispiel unten.

Im Menüpunkt 1 soll die Funktion **output** mit dem Bus als Parameter aufgerufen werden,

für den Menüpunkt 2 die Funktion **output_whole_line** mit dem Zeiger auf die erste Haltestelle als Parameter.

Im Menüpunkt 3 soll zuerst eine beliebige Anzahl n nächster anzufahrender Haltestellen eingegeben werden. Über eine Zählschleife von 1 bis n soll für jede aktuell simuliert angefahrne Haltestelle die Funktion **simulate_get_on_off** aufgerufen werden mit dem Bus und der aktuellen Haltestelle als Parameter sowie mit zwei erzeugten Zufallszahlen für jeweils 0 bis 9 aussteigende und 0 bis 9 einsteigende Fahrgäste. Anschließend soll die Funktion **output** für die aktuelle Haltestelle aufgerufen werden sowie der Bus simuliert zur nächsten Haltestelle in der Ringlinie fahren (also der Zeiger auf die aktuelle Haltestelle auf die nächste gesetzt werden)./

Write a **main** function in whose body, among other variables, a structure variable for a bus and two pointer variables to stops are defined, one for the first stop of the bus circular line and the second for the respective current stop at which the bus simulates stopping. Set

- the values of the components from the bus via a call to the **init** function and
- after a user guidance for inputting the number of stops, set both pointer variables to the returned pointer of your function **generate_circular_line**, i.e. the first stop in the generated bus circular line.

Via a menu selection, the tour of the bus from stop to stop is then to be simulated and outputs be made as in the example below.

In menu item 1 the function **output** is to be called with the bus as parameter,

for menu item 2 the function **output_whole_line** with the pointer to the first stop as parameter.

In menu item 3, an arbitrary number n of next stops to be approached is to be entered first. Using a counting loop from 1 to n for each currently simulated approached stop the function **simulate_get_on_off** is to be called with the bus and the current stop as parameters and two generated random numbers for 0 to 9 getting off and 0 to 9 getting on passengers respectively. Then the function **output** is to be called for the current stop as well as the bus is to drive simulated to the next stop in the circular line (i.e. set the pointer to the current stop to the next).

Beispiel Programmlauf/Example Program Run


```
Eingabe Anzahl Haltestellen / input number of bus stops: 5

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben  / show all bus stops
3 simuliere Stops             / simulate bus stops
>> 1
bus data: passengers: actual: 0 maximum in simulation: 0

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben  / show all bus stops
3 simuliere Stops             / simulate bus stops
>> 2
Bus Stop 1:      sums: off: 0    on: 0    stops: 0
Bus Stop 2:      sums: off: 0    on: 0    stops: 0
Bus Stop 3:      sums: off: 0    on: 0    stops: 0
Bus Stop 4:      sums: off: 0    on: 0    stops: 0
Bus Stop 5:      sums: off: 0    on: 0    stops: 0

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben  / show all bus stops
3 simuliere Stops             / simulate bus stops
>> 3
Eingabe Anzahl Stops / input number of stops: 1
simulate Bus Stop 1:      getting off: 0 getting on: 1
Bus Stop 1:      sums: off: 0    on: 1    stops: 1

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben  / show all bus stops
3 simuliere Stops             / simulate bus stops
>> 1
bus data: passengers: actual: 1 maximum in simulation: 1

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben  / show all bus stops
3 simuliere Stops             / simulate bus stops
>> 2
Bus Stop 1:      sums: off: 0    on: 1    stops: 1
Bus Stop 2:      sums: off: 0    on: 0    stops: 0
Bus Stop 3:      sums: off: 0    on: 0    stops: 0
Bus Stop 4:      sums: off: 0    on: 0    stops: 0
Bus Stop 5:      sums: off: 0    on: 0    stops: 0

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben  / show all bus stops
3 simuliere Stops             / simulate bus stops
>> 3
Eingabe Anzahl Stops / input number of stops: 1
simulate Bus Stop 2:      getting off: 0 getting on: 4
Bus Stop 2:      sums: off: 0    on: 4    stops: 1

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben  / show all bus stops
3 simuliere Stops             / simulate bus stops
>> 1
bus data: passengers: actual: 5 maximum in simulation: 5

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben  / show all bus stops
3 simuliere Stops             / simulate bus stops
>> 2
```

```

Bus Stop 1:      sums: off: 0    on: 1    stops: 1
Bus Stop 2:      sums: off: 0    on: 4    stops: 1
Bus Stop 3:      sums: off: 0    on: 0    stops: 0
Bus Stop 4:      sums: off: 0    on: 0    stops: 0
Bus Stop 5:      sums: off: 0    on: 0    stops: 0

Simulation Ring-Buslinie    / Simulation Bus Circular Line
0 Ende                  / end
1 Daten Bus anzeigen      / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops        / simulate bus stops
>> 3
Eingabe Anzahl Stops / input number of stops: 1
simulate Bus Stop 3:      getting off: 4 getting on: 4
Bus Stop 3:      sums: off: 4    on: 4    stops: 1

Simulation Ring-Buslinie    / Simulation Bus Circular Line
0 Ende                  / end
1 Daten Bus anzeigen      / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops        / simulate bus stops
>> 2
Bus Stop 1:      sums: off: 0    on: 1    stops: 1
Bus Stop 2:      sums: off: 0    on: 4    stops: 1
Bus Stop 3:      sums: off: 4    on: 4    stops: 1
Bus Stop 4:      sums: off: 0    on: 0    stops: 0
Bus Stop 5:      sums: off: 0    on: 0    stops: 0

Simulation Ring-Buslinie    / Simulation Bus Circular Line
0 Ende                  / end
1 Daten Bus anzeigen      / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops        / simulate bus stops
>> 1
bus data: passengers: actual: 5    maximum in simulation: 5

Simulation Ring-Buslinie    / Simulation Bus Circular Line
0 Ende                  / end
1 Daten Bus anzeigen      / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops        / simulate bus stops
>> 3
Eingabe Anzahl Stops / input number of stops: 3
simulate Bus Stop 4:      getting off: 3 getting on: 3
Bus Stop 4:      sums: off: 3    on: 3    stops: 1
simulate Bus Stop 5:      getting off: 4 getting on: 2
Bus Stop 5:      sums: off: 4    on: 2    stops: 1
simulate Bus Stop 1:      getting off: 0 getting on: 0
Bus Stop 1:      sums: off: 0    on: 1    stops: 2

Simulation Ring-Buslinie    / Simulation Bus Circular Line
0 Ende                  / end
1 Daten Bus anzeigen      / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops        / simulate bus stops
>> 1
bus data: passengers: actual: 3    maximum in simulation: 5

Simulation Ring-Buslinie    / Simulation Bus Circular Line
0 Ende                  / end
1 Daten Bus anzeigen      / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops        / simulate bus stops
>> 2
Bus Stop 1:      sums: off: 0    on: 1    stops: 2
Bus Stop 2:      sums: off: 0    on: 4    stops: 1
Bus Stop 3:      sums: off: 4    on: 4    stops: 1
Bus Stop 4:      sums: off: 3    on: 3    stops: 1
Bus Stop 5:      sums: off: 4    on: 2    stops: 1

Simulation Ring-Buslinie    / Simulation Bus Circular Line
0 Ende                  / end
1 Daten Bus anzeigen      / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops        / simulate bus stops
>> 3

```

```

Eingabe Anzahl Stops / input number of stops: 20
simulate Bus Stop 2:      getting off: 2 getting on: 1
Bus Stop 2:      sums: off: 2    on: 5    stops: 2
simulate Bus Stop 3:      getting off: 1 getting on: 1
Bus Stop 3:      sums: off: 5    on: 5    stops: 2
simulate Bus Stop 4:      getting off: 2 getting on: 0
Bus Stop 4:      sums: off: 5    on: 3    stops: 2
simulate Bus Stop 5:      getting off: 0 getting on: 2
Bus Stop 5:      sums: off: 4    on: 4    stops: 2
simulate Bus Stop 1:      getting off: 2 getting on: 1
Bus Stop 1:      sums: off: 2    on: 2    stops: 3
simulate Bus Stop 2:      getting off: 1 getting on: 2
Bus Stop 2:      sums: off: 3    on: 7    stops: 3
simulate Bus Stop 3:      getting off: 2 getting on: 2
Bus Stop 3:      sums: off: 7    on: 7    stops: 3
simulate Bus Stop 4:      getting off: 1 getting on: 1
Bus Stop 4:      sums: off: 6    on: 4    stops: 3
simulate Bus Stop 5:      getting off: 0 getting on: 3
Bus Stop 5:      sums: off: 4    on: 7    stops: 3
simulate Bus Stop 1:      getting off: 1 getting on: 2
Bus Stop 1:      sums: off: 3    on: 4    stops: 4
simulate Bus Stop 2:      getting off: 3 getting on: 1
Bus Stop 2:      sums: off: 6    on: 8    stops: 4
simulate Bus Stop 3:      getting off: 2 getting on: 4
Bus Stop 3:      sums: off: 9    on: 11   stops: 4
simulate Bus Stop 4:      getting off: 4 getting on: 2
Bus Stop 4:      sums: off: 10   on: 6    stops: 4
simulate Bus Stop 5:      getting off: 4 getting on: 0
Bus Stop 5:      sums: off: 8    on: 7    stops: 4
simulate Bus Stop 1:      getting off: 0 getting on: 3
Bus Stop 1:      sums: off: 3    on: 7    stops: 5
simulate Bus Stop 2:      getting off: 3 getting on: 2
Bus Stop 2:      sums: off: 9    on: 10   stops: 5
simulate Bus Stop 3:      getting off: 2 getting on: 3
Bus Stop 3:      sums: off: 11   on: 14   stops: 5
simulate Bus Stop 4:      getting off: 1 getting on: 1
Bus Stop 4:      sums: off: 11   on: 7    stops: 5
simulate Bus Stop 5:      getting off: 3 getting on: 3
Bus Stop 5:      sums: off: 11   on: 10   stops: 5
simulate Bus Stop 1:      getting off: 3 getting on: 2
Bus Stop 1:      sums: off: 6    on: 9    stops: 6

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops            / simulate bus stops
>> 1
bus data: passengers: actual: 2  maximum in simulation: 6

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops            / simulate bus stops
>> 2
Bus Stop 1:      sums: off: 6    on: 9    stops: 6
Bus Stop 2:      sums: off: 9    on: 10   stops: 5
Bus Stop 3:      sums: off: 11   on: 14   stops: 5
Bus Stop 4:      sums: off: 11   on: 7    stops: 5
Bus Stop 5:      sums: off: 11   on: 10   stops: 5

Simulation Ring-Buslinie      / Simulation Bus Circular Line
0 Ende                        / end
1 Daten Bus anzeigen          / show data of bus
2 alle Haltestellen ausgeben / show all bus stops
3 simuliere Stops            / simulate bus stops
>>

```

Last modified: Friday, 29 October 2021, 3:37 PM

