Using C++, write a program:

1. Define a C++ enumeration type (**enum struct**) with name **DaylightSaving** (Zeitumstellung). The enumeration shall contain the three enumeration values **summerTime**, **winterTime** and **unknown**.

2. Give the definition of a class **Clock** with following members:

* private variable with name hour **as** unsigned integer **.**
* private variable with name  minute as unsigned integer.
* private variable with name **daylightSaving** of above enumeration type **DaylightSaving**.
* public definition of a class for exception objects with name **HourTooBig**
* public definition of a class for exception objects with name **MinuteTooBig**
* public overloaded inline standard constructor initialising the clock(time) by an initialisation list to midnight **00:00** o'clock with unknown daylight saving.
* public overloaded inline constructor with three arguments for the to set hour, minute and daylight saving. If the argument for the hour is greater than 23 hours a new exception object **HourTooBig** shall be thrown. If the argument for the minute is greater than 59 minutes a new exception object **MinuteTooBig** shall be thrown.
* public definition of a member function with name **set\_daylightSaving** with a fitting parameter to set the value of the same named attribute.
* public definition of a member function with name **get\_daylightSaving** returning the value of the same named attribute.
* public definition of an overloaded unary increment operator **++** incrementing the clock by one minute.  
  Take care that at the full hour the minutes have to be set back from 59 to 0 and the hour needs to be increased by one as well as at 23:59 o'clock the clock has to be set back to 00:00 o'clock.
* public definition of a friend binary operator **==** to overload the comparison of the clock times for two given references to objects of type **Clock**.
* public declaration of a friend binary operator **<<** to overload the output function for a reference to an object of type **Clock** onto an output stream

3. Define outside of the class definition the binary output operator **<<** declared in last point of subtask 2. The clock time shall be outputted in form **hh:mm** always with five characters (e.g. **00:00**, **00:09**, **01:27**, **08:57**, **11:03**, **22:24**; see screen shots below).

4. Write a function **main** with following definitions and statements:

* Define a pointer of type **Clock** with name **clock1** to a newly defined object of class **Clock** on th heap with clock time **00:05** and **winterTime**.
* Define an object of type **Clock** with name **clock2** with clock time **23:09** and **summerTime**.
* Define a pointer of type **Clock** with name **clock3**.
* Set **clock1** to **summerTime**.
* Define a **for**-loop with exactly 120 iterations incrementing in its body **clock2**.
* Define a **try**-**catch**-block:  
  Try to input two unsigned integers for hour and minute values from standard character input stream. Assign to **clock3** a newly created object on heap with inputted clock time and unknown daylight saving. Test also in the try-block by using overloaded operator **==** whether **clock2** and **clock3** have same clock time - if yes string **"same time"** shall be outputted.  
  Catch possibly by the constructor thrown exception objects **HourTooBig** and **MinuteTooBig** (only these two, not all possible) and assign instead **clock3** a by the standard constructor newly created object.
* Write the times of **clock1**, **clock2** and **clock3** by using operator **<<** onto the standard character output stream