

Informatik für Mathematiker und Physiker **Serie 12** **HS07**URL: <http://www.ti.inf.ethz.ch/ew/courses/Info1.07/>**Skript-Aufgabe 100 (4 Punkte)**

Consider the following family of functions:

```
T foo (S i)
{
    return ++i;
}
```

with T being one of the types `int`, `int&` and `const int&`, and S being one of the types `int`, `const int`, `int&` and `const int&`. (This defines 12 different functions).

- Find the combinations of T and S for which the resulting function definition is syntactically valid, and explain your answer.
- Among the combinations found in a), find the combinations of T and S for which the resulting function definition is also semantically valid, meaning that function calls always have well-defined value and effect; explain your answer.
- For all combinations found in b), give precise postconditions for the corresponding function `foo`.

Skript-Aufgabe 109 (4 Punkte)

- Find all errors in the following program. Fix them and describe the functionality of the type `Clock`, by providing pre- and postconditions for the member functions.

```
1 #include <iostream>
2
3 class Clock {
4     Clock(unsigned int h, unsigned int m, unsigned int s);
5     void tick();
6     void time(unsigned int h, unsigned int m,
7              unsigned int s);
8 private:
9     unsigned int h_;
10    unsigned int m_;
11    unsigned int s_;
12 };
13
14 Clock::Clock(unsigned int& h,
15              unsigned int& m,
16              unsigned int& s)
17     : h_(h), m_(m), s_(s)
18 {}
19
```

```

20 void Clock::tick()
21 {
22     h_ += (m_ += (s_ += 1) / 60) / 60;
23     h_ %= 24; m_ %= 60; s_ %= 60;
24 }
25
26 void Clock::time(unsigned int& h,
27                 unsigned int& m,
28                 unsigned int& s)
29 {
30     h = h_;
31     m = m_;
32     s = s_;
33 }
34
35 int main() {
36     Clock c1 (23, 59, 58);
37     tick();
38
39     unsigned int h;
40     unsigned int m;
41     unsigned int s;
42     time(h, m, s);
43
44     std::cout << h << ":" << m << ":" << s << "\n";
45
46     return 0;
47 }

```

b) Implement an output operator for the class Clock.

Skript-Aufgabe 111 (4 Punkte)

Consider the generator `ansic` used in `choosing_numbers.C`. Since the modulus is $m = 2^{31}$, the internal computations of the generator will certainly overflow if 32 bits are used to represent `unsigned int` values. Despite this, the sequence of pseudorandom numbers computed by the generator is correct and coincides with its mathematical definition. Explain this!

Skript-Aufgabe 112 (4 Punkte)

Find a loaded dice that beats the fair dice in the game of choosing numbers. (This is a theory exercise.)

Challenge

You can solve the challenge exercise 113 from the lecture notes. It will be awarded a maximum of 8 points, and thus replaces two of the normal exercises.

Remark

This is a bonus sheet (for details see the course homepage). At the same time, this is the last regular exercise sheet. Number 13 will only contain the Schnellübung.

Abgabe: Bis 18. Dezember 2007, 15.15 Uhr.