Solution 1

a) The table looks as follows:

<table>
<thead>
<tr>
<th>Expression</th>
<th>Valid?</th>
<th>L- or R-value?</th>
<th>Value?</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i) (a=(b=5))</td>
<td>true</td>
<td>L-value</td>
<td>5</td>
</tr>
<tr>
<td>(ii) (1=a)</td>
<td>false</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(iii) ((a=5)*(b=7))</td>
<td>true</td>
<td>R-value</td>
<td>35</td>
</tr>
<tr>
<td>(iv) ((a=b)*(b=5))</td>
<td>true</td>
<td>R-value</td>
<td>unspecified</td>
</tr>
</tbody>
</table>

(ii) is not an expression, since the first operand of the assignment operator must be an lvalue, but 1 is a literal, hence an rvalue.

(iii) and (iv) are rvalues by definition of the binary multiplication operator. (i) is an lvalue by definition of the assignment operator.

(i) has value 5, obtained by assigning value 5 to \(b\) first (right assignment), and then to \(a\) (left assignment). (iii) has value 35, since the operands \((a=5)\) and \((b=7)\) have values 5 and 7, respectively.

In case of (iv), the value is unspecified. If the right operand is evaluated first, we get value 25, but if the left operand comes first, \(b\) may have some value other than 5, and the left operand evaluates to this other value. The final result will not be 25, then.

b) The computations work by applying the sum-formulas on pages 26 and 29 of lecture2p.pdf.
Solution 2

The output of the mixed number simply employs the properties of the / (division) and % (modulo) operators.

```cpp
#include <iostream>

int main () {
    // Input
    int fahrenheit;
    std::cin >> fahrenheit;

    // Computation
    const int ncelsius = 5 * (fahrenheit - 32); // numerator

    // Output as mixed number
    std::cout << fahrenheit << " degrees Fahrenheit are "
        << ncelsius / 9 << " " << ncelsius % 9
        << "/9 degrees Celsius."
    return 0;
}
```
Solution 3

```
// Informatik − Serie 2 − Aufgabe 3
// Programm: average_speed.cpp
// Autor: ... (Gruppe ...)
// Computes the average speed over a roundtrip A→B→A, given the ←
// average speeds
// for A→B and B→A

#include "tests.h" // remove slashes at beginning of line to test ←
or submit
#include <iostream>

int main()
{
    // inputs
    unsigned int x;
    std::cin >> x;
    unsigned int y;
    std::cin >> y;

    // computation and output
    std::cout << "Average speed from A to B and back is "
              << 2*x*y/(x+y) << " km/h.\n";
    return 0;
}
```