Exercise Week 05

GianAndrea Müller
mailto:muellegi@student.ethz

March 28, 2018

Time Schedule

- 20' Self assessment 2
- 15' Korrektur
- 10' Standard Library
- 15' Referenzen mit Übung

Learning Objectives

- Kenntnis von Bibliotheken
- Verständnis von Referenzen

Standardbibliotheken

```
1 #include <cmath>
2 #include <iostream>
3 using namespace std;
4
5 int main(){
6   cout << pow(3.3,6.5) << endl;
7   cout << sqrt(9.1) << endl;
8   cout << abs(-3.0) << endl;
9 }
```

Standardbibliotheken

```
1  #include <cmath>
2  #include <iostream>
3  using namespace std;
4
5  int main(){
6    cout << pow(3.3,6.5) << endl;
7    //Berechnet 3.3^6.5
8    cout << sqrt(9.1) << endl;
9    //Berechnet die Wurzel von 9.1
10   cout << abs(-3.0) << endl;
11   //Berechnet den Absolutbetrag von (-3)
12 }
```

Standardbibliotheken

```cpp
#include <iostream>
#include <cmath>
#include <cassert>
using namespace std;

int main(){
  double x;
  cin>>x; // try x = 2
  assert(x > 0);

  double sqrtx = sqrt(x);
  cout << abs(sqrtx*sqrtx - x) << "\n";

  return 0;
}
```

#include <cassert> #include <cmath>

Standardbibliotheken

```
1  bool in_circ_exp (double x, double y,
       double r)
2  {
3    return sqrt(x*x + y*y) < radius;
4  }
5
6  bool in_circ_cheap(double x, double y,
       double r)
7  {
8    return x*x + y*y < radius*radius;
9  }
```

Standardbibliotheken

```cpp
#include <iostream>
#include <algorithm>

int main(){
   cout << min(3.5,4.1) << "\n";
   cout << max(3.4,9.1) << "\n";
   return 0;
}
```

#include <algorithm>

Referenzen

```
1 void increment (int m) {
2 m++;
3 }
4 int main () {
5 int n = 3;
6 increment (n);
7 return 0;
8 }
```

Referenzen

```
1 int i = 1;
2 int& j = i;
3 i++; // i = 2
4 j++; // i = 3
```

Referenzen

```
1 void increment (int& m) { //only line
    changed
2 m++;
3 }
4 int main () {
5 int n = 3;
6 increment (n);
7 return 0;
8 }
```

Funktionstypen

- **Call by value**

```
1 bool even (unsigned int a){
2   while (a>=1) a-=2;
3   return a != 1;
4 }
```

- **Call by reference**

```
1 void half (int & b){
2   b /= 2;
3 }
```

Funktionstypen

```
1 // POST:  return value is the number of
2 //        distinct real solutions
3 //        of the equation ax^2+bx+c=0.
4 //        The solutions are written to s1
      and s2.
5
6 int solve_quadratic_equation (const double
      a, const double b, const double x,
      double& s1, double& s2);
```

Letzte Seite

assi-link