

I2-OOP sommer 2018 ordinær eksamen – løsningsforslag

OPGAVE 1

```
// MILESTONE 1 (Stang.h)
bool operator>(const Stang & left, const Stang & right);

// MILESTONE 1 (Stang.cpp)
bool operator>(const Stang & left, const Stang & right)
{
    return left.getLaengde() > right.getLaengde();
}
```

OPGAVE 2

```
// MILESTONE 2 (Plade.h)
template <class T>
class Plade
{
public:
    Plade(T laengde, T bredde, string unit);
    void setLaengde(T laengde);
    T getLaengde() const;
    void setBredde(T bredde);
    T getBredde() const;
    void print() const;
private:
    T laengde_;
    T bredde_;
    string unit_;
};

template <class T>
Plade<T>::Plade(T laengde, T bredde, string unit)
{
}

template <class T>
void Plade<T>::setLaengde(T laengde)
{
}

template <class T>
T Plade<T>::getLaengde() const
{
}

template <class T>
void Plade<T>::setBredde(T bredde)
{
}

template <class T>
T Plade<T>::getBredde() const
{
}

template <class T>
void Plade<T>::print() const
{
}
```

```
// MILESTONE 2 (test_Plade.cpp)
#include "Plade.h"

int main()
{
    Plade<int> p1(90, 21, "cm");
    Plade<double> p2(0.90, 0.21, "m");

    p1.print();
    p2.print();

    return 0;
}
```

OPGAVE 3

```
// MILESTONE 3 (Beholder.h)
#pragma once

class Beholder
{
public:
    Beholder();
    virtual ~Beholder();
    virtual double beregnVolumen() const = 0;
    virtual void print() const;
protected:
    double volumen_;
};
```

```
// MILESTONE 4 (Kasse.h)
#pragma once
#include "Beholder.h"

class Kasse : public Beholder
{
public:
    Kasse(double hoejde, double bredde, double dybde);
    virtual ~Kasse();
    virtual double beregnVolumen() const;
    virtual void print() const;
private:
    double hoejde_;
    double bredde_;
    double dybde_;
};
```

```
// MILESTONE 4 (Toende.h)
#pragma once
#include "Beholder.h"

class Toende : public Beholder
{
public:
    Toende(double hoejde, double bredde);
    virtual ~Toende();
    virtual double beregnVolumen() const;
    virtual void print() const;
private:
    double hoejde_;
    double radius_;
};
```

```

// MILESTONE 4 (test_Beholdere.h)
#include "Beholder.h"
#include "Kasse.h"
#include "Toende.h"

int main()
{
    Kasse k(50, 75, 40);
    Toende t(80, 20);

    k.print();
    t.print();

    return 0;
}

// MILESTONE 5 (Toende.h - ændringer)
public:
    Toende(double hoejde, double bredde, const char *indhold);
private:
    char * indhold_;

// MILESTONE 5 (Toende.cpp - ændringer)
Toende::Toende(double hoejde, double radius, const char * indhold)
{
    hoejde_ = (0 < hoejde && hoejde <= 100 ? hoejde : 0);
    radius_ = (0 < radius && radius <= 25 ? radius : 0);
    volumen_ = beregnVolumen();

    indhold_ = new char[strlen(indhold) + 1];

    strcpy(indhold_, indhold);
}

Toende::~~Toende()
{
    cout << "Destructoren for klassen Toende er kaldt" << endl;

    delete[] indhold_;
}

// MILESTONE 6 (test_Beholdere.cpp - tilføjelse)
#include <list>
#include <iterator>

using namespace std;

int main()
{
    Kasse k(50, 75, 40);
    Toende t(80, 20, "Olie");

    k.print();
    t.print();

    list<Beholder *> liste;

    liste.push_back(&k);
    liste.push_back(&t);

    list<Beholder *>::iterator i;

    for ( i = liste.begin(); i != liste.end(); i++)
        (*i)->print();

    return 0;
}

```