

Osa 3: Kotitehtävien vastauksia

Tehtävä 1, vastaus: Ohjelmassa tarvitsee kirjoittaa vain **main()** funktion koodi uudestaan. ***Luokkaan ja sen toteutukseen ei tarvitse koskea.***

```
#include "standardcalculator.h"
```

```
int main()
```

```
{  
    StandardCalculator *objectStandardCalculator;
```

```
  
    objectStandardCalculator = new StandardCalculator; // olio luodaan kekomuistiin
```

```
  
    objectStandardCalculator->sum(10,5); // olion kutsuessa luokan jäsenfunktiota käytetään pisteoperaattorin sijaan merkkejä ->
```

```
    objectStandardCalculator->subtraction(10,5);
```

```
    objectStandardCalculator->multiplication(10,5);
```

```
    objectStandardCalculator->division(10,5);
```

```
  
    delete objectStandardCalculator; // vapautetaan olion varaama tila
```

```
    objectStandardCalculator = nullptr; // siivotaan muisti
```

```
  
    return 0;
```

```
}
```

IN00BQ93 Laite- ja tuotesuunnittelun syventävät opinnot: ohjelmoinnin jatkokurssi

Osa 3: Kotitehtävien vastauksia

Tehtävä 2: Rakenna **AvoinVakioStaattinen** ohjelma siten, että ohjelmassa käytetään avoimia jäsenfunktioita, vakiofunktioita ja pinomuistia.

Tehtävän koodi seuraavalla sivulla.

Osa 3: Kotitehtävien vastauksia

```
#include <iostream>
using namespace std;

class MyClass
{
public:
    void setData(short paramValue) // tama on avoinfunktio, koska koodi on luokan rakenteessa
    {
        myVariable = paramValue;
        myStaticVariable=10;
        this->myStaticFunction(); // voit kutsua funktiota myos ilman this osoitinta
    }
    void showData() const // tama on seka vakio- etta avoinfunktio
    {
        cout << "myVariable= " << myVariable << endl;
    }
    static void myStaticFunction() // staattinen jassenfunktio
    {
        myStaticVariable++;
        cout << "Luokan staattisen muuttujan arvo= " << myStaticVariable << endl;
    }

private:
    short myVariable;
    static short myStaticVariable;
};
```



```
short MyClass::myStaticVariable = 0;

int main()
{
    MyClass objectMyClass;
    objectMyClass.setData(10);
    objectMyClass.showData();
    objectMyClass.myStaticFunction();

    MyClass objectMyClass2; // luodaan toinen olio.
    objectMyClass2.myStaticFunction(); // kutsutaan staattista jassenfunktioita

    return 0;
}
```