

# PRAKTIKUM 3

## KELAS DAN OBJEK LANJUTAN

### 1. Listing Program 1

```
#include <iostream>

class CONTOH {
    int X;
public:
    void SetX(int XX) {
        X = XX;
    }
    int GetX() {
        return X;
    }
};

int Kuadrat(CONTOH A, int N) {
    A.SetX(N);
    return (A.GetX() * A.GetX());
}

int main() {

    CONTOH O;

    // Memanggil fungsi Kuadrat
    cout<<"Kuadrat dari 10 adalah : "<<Kuadrat(O,10);

    return 0;
}

class CONTOH {
    int X;
public:
    void SetX(int XX) {
        X = XX;
    }
    int GetX() {
        return X;
    }
};

int Kuadrat(CONTOH A, int N) {
    A.SetX(N);
    return (A.GetX() * A.GetX());
}

int main() {

    CONTOH O;

    // Memanggil fungsi Kuadrat
    cout<<"Kuadrat dari 10 adalah : "<<Kuadrat(O,10);

    return 0;
}
```

## 2. Listing Program 2

```
#include <iostream>

class CONTOH {
    int X;
public:
    void SetX(int XX) {
        X = XX;
    }
    int GetX() {
        return X;
    }
};

CONTOH MyFunc() {
    CONTOH A;

    A.SetX(30);

    return A; // Mengembalikan nilai yang berupa objek
}

int main() {

    CONTOH O;

    // Memanggil fungsi Kali2 dan nilainya dimasukkan
    // ke dalam objek O
    O = MyFunc();

    cout<<"Nilai X di dalam O adalah : "<<O.GetX();

    return 0;
}
```

## 3. Listing Program 3

```
#include <iostream>

class CONTOH{
    int X, Y;
public:
    void SetXY(int XX, int YY) {
        X = XX;
        Y = YY;
    }
    friend int KALI(CONTOH A);
};

// Mendefinisikan fungsi KALI yang bukan termasuk member function
// dari kelas CONTOH

int KALI(CONTOH A) {
    // Mengakses bagian private secara langsung dari kelas CONTOH
    return (A.X * A.Y);
}

// Fungsi utama
```

```

int main() {

    // Melakukan instansiasi kelas CONTOH
    CONTOH O;

    O.SetXY(20, 3); // Mengeset nilai X=20 dan Y=3

    cout<<"Hasil kali : "<<KALI(O);

    return 0;
}

```

#### **4. Listing Program 4**

```

#include <iostream>

class KESATU {
    int X, Y;
public:
    KESATU(int XX, int YY) {
        X = XX;
        Y = YY;
    }
    friend class KEDUA; // Mendeklarasikan sebuah friend class
};

// Definisi dari kelas KEDUA
class KEDUA {
    // ...
public:
    int Kali(KESATU A);
    // ...
};
int KEDUA::Kali(KESATU A) {
    return (A.X * A.Y);
};

int main() {

    // Melakukan instansiasi dari kelas KESATU
    KESATU O(40, 3);

    // Melakukan instansiasi dari kelas KEDUA
    KEDUA P;

    // Memanggil fungsi Kali yang terdapat pada kelas KEDUA
    cout<<"Hasil kali : "<<P.Kali(O);

    return 0;
}

```