

Grupa B

1. Co to jest const + przykładowe fragmenty kodu.
2. Co to jest klasa abstrakcyjna, z jakim pojęciem jest ściśle związana, do czego służy?
3. Uzupełnij i uzasadnij. Coś z przeładowaniem operatora << i vectorem. Nie pamiętam kodu.
4. Jaka będzie kolejność wywoływania destruktorów, uzasadnij

Kod:

```
class Baza1 {...};
class Baza2 {...};
class Baza3 {...};
class x: public Baza1, public Baza2 {...};
class y: private Baza3, public X, public Baza1 {...};
```

```
int main()
{Y a;}
```

5. Uzupełnij i uzasadnij

Kod:

```
//uzupełnij
struct box{
//uzupełnij
    T a;
};
```

```
int main(){
    box<> b1 = 3.14;
    box<unsigned> b2 = 13;
    double d = b1;
    unsigned u = b2;
}
```

Grupa A

1. static
2. Co to jest dziedziczenie i jakie są jego rodzaje, przykład.
3. najprawdopodobniej to samo co w grupie B
4. Jaka będzie kolejność wywoływania konstruktorów, uzasadnij

Kod:

```
class Baza1 {...};
class Baza2 {...};
class Baza3 {...};
class x: public Baza1, public Baza2 {...};
class y: private Baza3, public X, public Baza1 {...};
```

```
int main()
```

```
{Y a;}
```

5. najprawdopodobniej to samo co w grupie B

Dołączył: 15 Maj 2008

Posty: 4

□ Wysłany: 2009-01-29, 11:39



Co do 5 w grupie A, powinno to wyglądać mniej więcej tak:

Kod:

```
template<typename T=int>
struct box {
    box(T x) : a(x) {}
    operator T(){ return (T)a; }
```

```
    T a;
};
```

```
int main()
{
    box<> b1 = 1;
    box<double> b2 = 2.2;
    double d = b2;
    int i = b1;
}
```

Dla grupy B analogicznie.

////////////////////////////////////

Zad 3. gr A:

(chyba dobrze zapamiętałem)

Kod:

// uzupełnij

```
int main()
{
    vector<int> veci;
    vector<double> vecd;

    for(int i = 0; i < 10; ++i){
```

```
        veci.push_back(i);
        vecd.push_back(i);
    }
    vecd = veci;

    cout << veci << endl << vecd << endl;
}
```

Program trzeba było uzupełnić lub zmodyfikować tak, aby wyświetlał:

Kod:

```
0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
```