

1. Opisz `dynamic_cast`, zaprezentuj przykłady kodu.
2. Kontenery z STL'a (charakterystyka, podział, kwestie bezpieczeństwa itp.)
3. Uzupełnij kod tak aby działał, odpowiedź uzasadnij.

Kod:

```
/*uzupelnij*/

int main(){
    //tu stworzone zostaly obiekty normal_order i reverse_order, ale nie pamietam jak to dokladnie
    wygladalo
    typedef /*uzupelnij*/ kontener_typ;
    kontener_typ c(normal_order);
    //tu byl kod ktory wstawial jakies 4 trzy-literowe stringi do c uzywajac insert
    copy(c.begin(), c.end(), std::ostream_iterator<std::string>(std::cout, ", "));
    kontener_typ d(reverse_order);
    //tu byl jakis kod
    copy(c.begin(), c.end(), /*uzupelnij*/);
    copy(d.begin(), d.end(), std::ostream_iterator<std::string>(std::cout, ", "));
    //tu byl jakis kod
}

//program mial wypisac te 4 stringi w porzadku alfabetycznym a potem odwrotnym
```

4. Uzupełnij kod tak aby klasa `NonInheritable` stała się klasą finalną (czyli taka po której nie można dziedziczyć). Odpowiedź uzasadnij.

Kod:

```
class NonInheritable;
/*uzupelnij*/

//class Pochodna : public NonInheritable{};

int main(){
    NonInheritable a;
    //Pochodna b; //odkomentowanie tej linii ma spowodowac blad
}
```