

Zadanie 13a

Zaprojektować i zaimplementować algorytm ogólny *match()* realizujący porównywanie wzajemne dwóch sekwencji szukając pierwszej pary zgodnych elementów (porównaj algorytm *mismatch()* z biblioteki standardowej). Funkcja ma zwrócić iteratory elementów zgodnych. Jeżeli w sekwencjach nie ma elementów zgodnych funkcja ma zwrócić iterator końca pierwszej sekwencji oraz odpowiadający mu iterator sekwencji drugiej.

Przewidzieć możliwość dostarczenia przez użytkownika funkcji orzekającej o zgodności elementów sekwencji.

Przykład kodu testującego

```
int t[] = {3, 8, 2, 9, 2, 7, 2, 5, 8};
std::vector<int> v(t, t+4);
std::list<int> l(t+4, t+9);
std::pair<std::vector<int>::iterator,
         std::list<int>::iterator> p;
p = match(v.begin(), v.end(), l.begin());
std::cout << *p.first << ' ' << *p.second << '\n';
```

Zadanie 13b

Zaprojektować i zaimplementować algorytm ogólny *for_each()* w wersji przyjmującej elementy pochodzące z dwóch sekwencji (porównaj algorytm *for_each()* i *transform()* z biblioteki standardowej).