
laboratorium: zadanie 12 (dodatkowe)

termin: 9–11 czerwca 2015 r.

KURS JĘZYKA C++

OBIEKTY FUNKCYJNE I LAMBDA

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

Zadanie 1a.

Zdefiniuj klasę `osoba` z polami typu `string` do zapamiętania imienia i nazwiska (pola `imie` i `nazwisko`) oraz z polem typu `time_point` do zapamiętania daty urodzenia (pole `ur`). W klasie tej powinien być zdefiniowany konstruktor do zainicjalizowania wszystkich wymienionych pól obiektu wartościami przekazanymi przez argumenty. Zaimplementuj też pełną semantykę kopiowania w klasie `osoba`.

Stwórz plik tekstowy z danymi osobowymi, którymi będzie można zainicjalizować kolekcję osób: w pierwszej linii liczba danych a w kolejnych informacje personalne (imię, nazwisko, rok, miesiąc i dzień urodzenia). Na przykład:

```
4
Jan Kowalski 1970 4 3
Anna Nowak 1981 11 17
Monika Walas 1965 10 30
Bartosz Rybak 2003 2 21
```

Zadanie 1b.

Napisz program, który wczyta dane z pliku z danymi osobowymi do pamięci, umieszczając wszystkie informacje w kolekcji `vector<osoba>`. Następnie posortuj te dane najpierw względem nazwiska i imienia a potem względem daty urodzenia (po każdym sortowaniu wypisz wszystkie osoby z kolekcji). Do sortowania użyj obiektów funkcyjnych.

Zadanie 1c.

Napisz program, który wczyta dane z pliku z danymi osobowymi do pamięci, umieszczając wszystkie informacje w kolekcji `list<osoba>`. Następnie zastosuj do tej kolekcji algorytm `for_each` z obiektem funkcyjnym z takim stanem wewnętrznym, który pozwoli potem na wskazanie osoby najmłodszej i najstarszej w tej kolekcji.

Zadanie 2a.

Do kolekcji `deque<int>` wpisz kilkadziesiąt losowo wygenerowanych liczb — liczby mają być wygenerowane z rozkładem jednostajnym i mają pochodzić ze zbioru $\{-10, \dots, 10\}$.

Następnie zdefiniuj predykat, który sprawdzi, czy podana wartość jest dodatnia i parzysta. Predykat ten skonstruuj z predefiniowanych obiektów funkcyjnych. Predykatu tego użyj do zliczenia wszystkich elementów kolekcji spełniających podany warunek (algorytm `count_if`) oraz do wyznaczenia pierwszej wartości spełniającej ten warunek (algorytm `find_if`).

Zadanie 2b.

Do kolekcji `deque<double>` wpisz kilkadziesiąt losowo wygenerowanych liczb — liczby mają być wygenerowane z rozkładem normalnym i mają pochodzić z przedziału $[0, 10]$.

Następnie usuń z tej kolekcji algorytmem `remove_if` wszystkie wartości mniejsze od 2.0 i większe od 8.0. W algorytmie tym wykorzystaj odpowiednio zdefiniowaną predykatową funkcję `lambda`.

Na koniec za pomocą algorytmu `count_if` zlicz wszystkie elementy w kolekcji, które są większe od 4 i mniejsze od 6. W algorytmie tym wykorzystaj odpowiednio zdefiniowaną predykatową funkcję `lambda`.