

ELEMENTY JĘZYKA C++: tablice, pojemnik STL `<vector>`, generator liczb losowych.

**1. Orzeł czy reszka?**

Napisz program symulujący rzut monetą. Uruchom go wiele razy (tzn. wiele losowań w pętli). Czy uzyskane wyniki wyglądają Twoim zdaniem na losowe?

**2. Element najmniejszy**

Napisz program znajdujący indeks najmniejszego elementu tablicy. Program powinien wczytywać długość tablicy, tworzyć tę tablicę w pamięci, wypełniać losowymi liczbami rzeczywistymi z przedziału od 0 do 1, a następnie wypisywać elementy tablicy wraz z indeksami oraz znaleziony indeks i wartość elementu najmniejszego.

**3. Sito Eratostenesa**

Napisz program znajdujący metodą sita Eratostenesa wszystkie liczby pierwsze mniejsze od danej liczby naturalnej. Program powinien czytać tę liczbę ze standardowego wejścia, a wynik wypisywać na standardowe wyjście.

**4. Sortowanie bąbelkowe**

Napisz program sortujący metodą bąbelkową tablicę wczytaną przez użytkownika. Porównujemy najpierw dwa pierwsze elementy i jeśli są w złej kolejności, to zamieniamy je. Następnie robimy to samo z drugim i trzecim elementem, itd. aż do końca tablicy. Jeżeli w takim pojedynczym przebiegu wszystkie pary były w dobrej kolejności to znaczy, że tablica jest już posortowana. Jeśli natomiast musieliśmy wykonać przynajmniej jedną zamianę, to powtarzamy całą procedurę od początku.

**5. Obliczenia statystyczne**

Niech dana będzie próba losowa  $N$  wartości  $x_i$ . Średnia i odchylenie standardowe z tej próby wynoszą odpowiednio:

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i \quad \sigma_x = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)^2}$$

Napisz program, który pobiera ze standardowego wejścia dodatnie liczby rzeczywiste aż do momentu, gdy użytkownik wpisze wartość  $-1$ , a następnie wyznacza średnią i odchylenie standardowe podanych liczb. *Wskazówka:* użyj pojemnika `<vector>` ze standardowej biblioteki szablonów STL.

**6. Dominanta sondażowa**

Dominantą zbioru danych w statystyce nazywamy taką wartość, która występuje w nim najczęściej. Napisz kod, który przetwarza tablicę danych sondażowych, aby ustalić ich dominantę. Odpowiedź na pytanie sondażowe polegała na podawaniu liczby z zakresu od 1 do 10. W sytuacji gdy istnieje wiele dominant, można wybrać dowolną z nich. Program powinien wczytywać odpowiedzi na pytanie ankietowe do momentu, w którym użytkownik wprowadzi wartość 0.

**7. Kostka sześcienna**

Napisz program symulujący serię rzutów kostką sześcienną. Niech program prosi o podanie liczby rzutów, a następnie stworzy histogram rezultatów losowania. Zadbaj o niedeterministyczne zachowanie programu.

**8. Wicelider**

Napisz program znajdujący położenie drugiego co do wielkości elementu tablicy. Program powinien wypełniać tablicę o zadanej długości losowymi liczbami rzeczywistymi z przedziału od 0 do 1, a następnie wypisywać elementy tablicy wraz z indeksami oraz znalezionej indeks i wartość elementu drugiego co do wartości.

**9. Losowanie Lotto \***

Napisz program losujący 6 parami różnych liczb naturalnych z przedziału od 1 do 49 włącznie i wypisujący je w kolejności rosnącej. W programie nie używaj sortowania. *Wskazówka:* użyj pojemnika `<set>` ze standardowej biblioteki szablonów STL.

**10. Sortowanie – do wyboru, do koloru**

Sortowanie bąbelkowe jest najgorszym znanym algorytmem sortowania, jednakże prostym w implementacji. Istnieją również inne (lepsz) algorytmy, wśród nich: sortowanie przez wstawianie i sortowanie przez wybór. Napisz program sortujący tablicę liczb losowych jedną z powyższych metod, jak dla sortowania bąbelkowego.

**11. Szyfr podstawieniowy**

Napisz program używający szyfru podstawieniowego, w którym wszystkie wiadomości składają się z wielkich liter i znaków interpunkcyjnych. Pierwotna wiadomość jest zwana tekstem jawnym, zaś szyfrogram tworzy się poprzez podmianę każdej z liter na inną. Utwórz w programie tablicę typu `const` składającą się z 26 elementów `char` służących do szyfrowania. Program powinien odczytywać tekst jawny i wyprowadzać odpowiadający mu szyfrogram.

**12. Szyfr podstawieniowy 2 \***

Zmodyfikuj powyższy program w taki sposób, by konwertował szyfrogram z powrotem na tekst jawny w celu zweryfikowania kodowania i dekodowania.

**13. Szyfr podstawieniowy 3 \*\***

Aby jeszcze bardziej utrudnić problem szyfru podstawieniowego, zmodyfikuj program w taki sposób, by zamiast wykorzystywać wbudowaną tablicę wartości `const`, generował w sposób losowy wzorec szyfrowania. W praktyce oznacza to umieszczenie losowych znaków w każdym elemencie tablicy. Pamiętaj, że dana litera nie może być substytutem samej siebie. Nie możesz także użyć tej samej litery dwukrotnie.

Pytania, a także rozwiązania zadań, można wysyłać na adres: MDABROWSKI@FUW.EDU.PL.