

ELEMENTY JĘZYKA C++: pętle (for, while, do-while), biblioteka `<cmath>`.

1. Iteracyjna silnia

Napisz program, który wczytuje ze standardowego wejścia liczbę naturalną i wypisuje na standardowe wyjście jej silnię. Pamiętaj, że $0! = 1$.

2. Suma narastająca

Napisz program obliczający sumę narastającą liczb wprowadzanych przez użytkownika, który zakończy swoje działanie, gdy użytkownik wprowadzi 0.

3. Trójkąty połączone bokami

Napisz program, który wykorzystuje tylko dwa polecenia wyjściowe *cout*, aby wygenerować wzór złożony z symboli kratki # ułożonych w kształt dwóch trójkątów o boku N połączonych ze sobą bokami.

4. Łamacz haseł

Napisz program służący do weryfikacji haseł, który pobiera od użytkownika login i hasło aż do momentu, gdy wpisane dane umożliwią dostęp do systemu.

5. Liczby pierwsze

Napisz program sprawdzający, czy dana liczba naturalna jest pierwsza. Program powinien wczytywać liczbę ze standardowego wejścia i drukować na standardowe wyjście odpowiedni komunikat.

6. Sumowanie szeregu – funkcja cos

Funkcję cos można przedstawić w postaci szeregu:

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots$$

Napisz program obliczający wartość funkcji cos dla argumentu x wczytywanego z klawiatury. Obliczanie powinno odbywać się poprzez sumowanie szeregu. Sumowanie powinno zakończyć się, gdy kolejny wyraz sumy ma moduł mniejszy niż wybrana precyzja obliczeń.

7. Ankieter *

Napisz program udostępniający opcję sumowania wyników ankiety, w której mogą wystąpić trzy różne wartości. Dane wejściowe wprowadzane do programu to pytanie ankietowe oraz trzy możliwe odpowiedzi. Pierwszej odpowiedzi przypisywana jest wartość 1, drugiej 2, a trzeciej 3. Odpowiedzi są sumowane do chwili, w której użytkownik wprowadzi 0 – wtedy program powinien pokazać wyniki ankiety. Postaraj się wygenerować wykres słupkowy, pokazujący wyniki przeskalowane w taki sposób, aby zmieściły się na ekranie bez względu na liczbę udzielonych odpowiedzi.

8. Projektant wzorów #

Wymyśl symetryczny kształt składający się ze znaków kratki, a następnie spróbuj napisać program, który go generuje, stosując się do znanego Ci ograniczenia dotyczącego tworzenia wzorów (zezwalającego na wykorzystanie tylko dwóch poleceń wyjściowych – jednego, wyświetlającego znak kratki, i drugiego, wyprowadzającego znak końca wiersza).

9. Łamacz haseł 2

Zmodyfikuj program służący do weryfikacji haseł tak, aby dawał użytkownikowi tylko kilka szans na podanie poprawnego hasła (użycie łamacza haseł będzie trudne).

10. Rozkład na czynniki pierwsze

Napisz program rozkładający daną liczbę naturalną na czynniki pierwsze w następujący sposób. Sprawdzamy, czy liczba dzieli się przez 2. Jeżeli tak, to stwierdzamy, że dwójka występuje w jej rozkładzie na czynniki pierwsze, a samą liczbę dzielimy przez 2. Czynność tę powtarzamy, aż liczba przestanie być podzielna przez 2. Następnie powtarzamy tę samą procedurę badając podzielność przez 3, 4, itd., aż rozważana liczba stanie się równa 1.

11. Sumowanie szeregu – inne funkcje

Napisz program obliczający wartość innych funkcji (np. \sin , \exp , \ln) metodą sumowania odpowiedniego szeregu. Sumowanie powinno zakończyć się po zsumowaniu liczby wyrazów wczytanej przez użytkownika. Zbadaj, ile wyrazów szeregu należy sumować, aby uzyskać dokładność wyniku na poziomie 1%.

12. Arytmetyka binarna

Jeśli znasz arytmetykę binarną i wiesz, w jaki sposób można zamieniać liczby dziesiętne na binarne (i odwrotnie), spróbuj napisać programy, które wykonują takie konwersje dla liczb o nieograniczonej długości (możesz założyć, że są na tyle małe, iż mogą być przechowywane w standardowym typie *int* języka C++).

13. Liczby heksadecymalne *

Czy znasz liczby heksadecymalne? Napisz program, który pozwoli użytkownikowi na podanie liczby w systemie binarnym, dziesiętnym lub heksadecymalnym, a następnie wyświetli ją w każdym z nich.

14. Konwerter systemów liczbowych **

Potrzebujesz dodatkowego wyzwania? Uogólnij kod poprzedniego zadania w taki sposób, aby stworzyć program, który przekształca dowolną liczbę z systemu szesnastkowego lub niższego na liczbę opartą na innym systemie.

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