

Laboratoria nr 3

Tablice i pętle w C

Przed uczestnictwem w zajęciach student powinien:

- znać pętle: `while`, `for`, `do-while`,
- znać instrukcję wyboru `switch`,
- znać instrukcje `break` i `continue`,
- znać wskaźniki i adresy,
- znać mechanizm rzutowania,
- umieć deklarować tablice wielowymiarowe i tablice wskaźników.

Materiały

- http://www.cs.put.poznan.pl/wcomplak/BFILES/C_W_1.PDF
- http://www.cs.put.poznan.pl/wcomplak/BFILES/C_W_2.PDF
- http://www.cs.put.poznan.pl/wcomplak/BFILES/C_W_4.PDF

Zadania

1. Odwracanie tablicy jednowymiarowej.
2. Znajdowanie największej wartości wyrazu ciągu Fibonacciego nie większej od danej liczby podanej z wejścia (iteracyjnie).
3. Obliczanie NWD dla 2 wejściowych liczb naturalnych.
4. Obliczanie NWW dla 2 wejściowych liczb naturalnych.
5. Liczba doskonała to liczba równa sumie swoich dzielników mniejszych od niej samej. Napisz program znajdujący wszystkie liczby doskonałe w przedziale $<1, a>$, np.: $28 = 1 + 2 + 4 + 7 + 14$.
6. Liczby zaprzyjaźnione to takie dwie liczby, z których każda jest równa sumie dzielników drugiej liczby mniejszych od tej liczby. Na przykład 220 i 284 są zaprzyjaźnione, ponieważ: 220 dzieli się przez 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 i 110, których suma wynosi 284, zaś 284 dzieli się przez 1, 2, 71 i 142, których suma wynosi 220. Napisz program znajdujący wszystkie liczby zaprzyjaźnione z przedziału $[1, a]$.
7. Liczba pierwsza N to taka, która dzieli się tylko przez 1 i N . Wyznaczenie wszystkich liczb pierwszych w przedziale $<A, B>$ algorytmem sita Erastotenesa.
8. Sprawdzić prawdziwość twierdzenia, że każda liczba parzysta jest sumą dwóch liczb pierwszych.
9. Liczby bliźniacze to liczby pierwsze, których różnica wynosi dwa, a więc np.: 3 - 5, 11 - 13. Znajdź wszystkie pary liczb bliźniaczych z przedziału $<1, a>$.
10. Liczba Mersenne'a to liczba pierwsza postaci $2^p - 1$, przy czym p samo jest liczbą pierwszą. Napisz program znajdujący takie liczby w przedziale $<1, a>$.
11. Każda liczba całkowita może być jednoznacznie rozłożona na iloczyn potęg liczb pierwszych. Napisz program, który dla danej liczby całkowitej dokonuje rozkładu.
12. Napisać funkcję potęgowania liczby naturalnej za pomocą mechanizmu wielokrotnego dodawania (iteracyjnie).
13. Napisz program wyznaczający resztę z dzielenia liczby naturalnej z wykorzystaniem mechanizmu wielokrotnego odejmowania.
14. Napisać funkcję wyznaczania silni liczby naturalnej n (iteracyjnie).

15. Hipoteza Simmonsa mówi, że tylko 4 silnie można wyrazić jako iloczyn trzech kolejnych liczb całkowitych. Oto jedna z nich: $4! = 2 * 3 * 4$. Znajdź trzy pozostałe. Czy możesz ich znaleźć więcej i obalić hipotezę?
16. Obliczyć ciąg symboli Newtona dla $i = 1 \dots k$, $(n \text{ po } i) = n! / (i! * (n - i)!)$ dla podanych liczb n, k .
17. Napisz program wyszukiwania w podanym przedziale $\langle D, G \rangle$ wszystkich liczb naturalnych (pitagoryjskich) a, b , dla których istnieje liczba naturalna c spełniająca warunek $a^2 + b^2 = c^2$.
18. Dla danej macierzy wyznaczyć sumę liczb na głównej przekątnej.
19. Dla danej macierzy wyznaczyć stosunek sumy wartości liczb nad główną przekątną do sumy wartości liczb pod główną przekątną.
20. Mnożenie dwóch macierzy.
21. Napisz program czytania n i rysowania macierzy o rozmiarach $n \times n$ typu lewa-dolna/prawa-górna. Macierz lewa-dolna zawiera na przekątnej głównej i pod tą przekątną znaki X, natomiast powyżej przekątnej głównej są znaki spacji.
22. Dla danej macierzy wejściowej napisać program sprawdzający czy jest ona symetryczna, czy zawiera puste kolumny (tzn. wypełnione zerami).
23. Dla danej macierzy wejściowej wyznaczyć stosunek sum wartości dwóch kolumn lub wierszy (decyduje użytkownik), przy czym numery tych kolumn/wierszy są również danymi wejściowymi.
24. Zliczyć ilość wystąpień wzorca w ciągu bazowym (oba są ciągami znakowymi).
25. Wypisać znak, który występuje największą ilość razy w znakowym ciągu wejściowym.
26. Wyznaczyć najdłuższy podciąg, wejściowego ciągu znakowego składający się z tych samych znaków zdefiniowanych na wejściu.
27. Sortowanie bąbelkowe.
28. Sortowanie przez wybieranie.
29. Sortowanie szybkie.