

Python – zadania

Język Python

1. Napisz funkcję obliczającą pierwiastki równania kwadratowego i przetestuj jej działanie. Wynik powinien być zwracany w postaci krotki lub listy. Przetestuj przekazując różne wartości a,b,c do funkcji i wyświetlając wyniki. Ustaw domyślne wartości b i c na 0.
 - pusta krotka - ()
 - krotka z jednym elementem, np. liczbą 1.0 - (1.,)
2. Napisz program – grę polegającą na odgadywaniu wylosowanej liczby. Program powinien pobierać liczbę wprowadzoną przez użytkownika i w przypadku nieodgadnięcia wyświetlić informację, czy wprowadzona liczba jest większa, mniejsza od wylosowanej. Powinny być liczone próby odgadnięcia liczby.
3. Cząsteczki DNA składają się z 4 zasad oznaczonych jako A, T, G, C, przy czym A łączy się z T, a G łączy się z C. Napisz program, który dla podanego ciągu znaków zbudowanych z liter ATGC, np. „TTAGTAGGGTTGTCCAT”:
 - a. Obliczy długość ciągu
 - b. Sprawdzi poprawność ciągu (czy występują odpowiednie litery: A, T, G, C)
 - c. Wygeneruje ciąg komplementarny, czyli zbudowany poprzez odpowiednie połączenie zasad A-T i G-C, np. dla ciągu „ATCGAA” utworzony będzie ciąg „TAGCTT”.
 - d. Odwróci sekwencję ciągu
 - e. Policz liczbę każdej z zasad
 - f. Policz liczbę wystąpień sekwencji „GG”
 - g. Policz liczbę sekwencji „ATA”

Kolejne elementy programu najlepiej umieścić w osobnych funkcjach.

Biblioteka numpy

1. Utwórz losową macierz o wymiarach 10x10. Znajdź największy i najmniejszy element.
2. Wypisz największe elementy w każdym z wierszy i każdej z kolumn.
3. Utwórz macierz o wymiarach 5x5 wypełnioną początkowo zerami. Na każdym brzegu macierzy ustaw jedynki (góra, dół, lewo, prawo). Napisz funkcję zamieniającą zera na jedynki i odwrotnie.
4. Utwórz dwie macierze wypełnione losowymi wartościami o takich samych wymiarach 5x5. Wykonaj na nich operacje:
 - a. Dodawania
 - b. Odejmowania
 - c. Mnożenia macierzy
 - d. Pomnożenia przez 2 i podzielenia przez 10
 - e. Wypisania wartości znajdujących się pośrodku macierzy
 - f. Zamień macierz na wektor