

Numer indeksu	Numer zadania
21286	1
21289	2
21292	3
21293	4
21294	5
21295	6
21297	7
21298	8
21299	9
21300	10
21301	11
21302	12
21303	13
21305	14
21306	15
21307	16
21310	17
21406	18
21617	19
21618	20

LISTA ZADAŃ

1. Napisz program symulujący grę w statki. Wygeneruj tablicę 10x10 i na początek wypełnij ją zerami. Napisz funkcję, która wypełni tablicę losowo umieszczonymi 10 jednomasztowcami wpisując w pole tablicy wartość 1 (dla uproszczenia założmy, że jednomasztowce mogą również być umieszczone obok siebie). Tablica ta jest nieznana użytkownikowi. Użytkownik ma 10 strzałów, w których podaje numer wiersza (od 1 do 10) oraz numer kolumny (od 1 do 10). Jeśli trafi w pole z jednomasztowcem (jedyneką) wypisz komunikat „Trafiony! Zatopiony!”. Jeśli strzeli w pole, które już wcześniej strzelał, wyświetl komunikat „Już tu strzelałeś/strzelałaś” (możesz zmienić wartość pola, w które trafił użytkownik wcześniej na inną liczbę, np. 2). Jeśli użytkownik trafi w pole o wartości 0 wypisz „Pudło”. Na koniec wyświetl komunikat pokazujący ile statków udało się zatopić.
2. Napisz program symulujący grę „Jednoręki bandyta”. Wygeneruj tablicę 3x3 i na początek wypełnij ją zerami. Napisz funkcję, która losowo wypełni tablicę liczbami od 0 do 9. Gracz ma 10 prób i wygrywa, jeśli w środkowym wierszu wszystkie cyfry są takie same. Jeśli będzie wygrana to wyświetl tablicę oraz komunikat „Wygrana!”. Na koniec wyświetl komunikat pokazujący ile razy udało się wygrać.
3. Napisz program symulujący grę „papier-nożyce-kamień”. Niech gra toczy się do trzech wygranych. Kolejno potem następują rundy, w których może wygrać komputer, użytkownik lub może być remis. Na koniec wyświetl informacje kto wygrał grę i jakie były końcowe wyniki obu graczy.
4. Napisz program symulujący grę w Dużego Lotka. Użytkownik wprowadza 6 liczb z zakresu od 1 do 49. Następnie komputer generuje 6 losowych cyfr z tego samego zakresu i wyświetla je w kolejności rosnącej. Wyświetl odpowiednie komunikaty dla następujących trafionych liczb:
 - Dla dwóch – „Wygrałeś 2 zł”,
 - Dla trzech – „Wygrałeś 500 zł”,
 - Dla czterech – „Wygrałeś 20 000 zł”,
 - Dla pięciu – „Wygrałeś 250 000 zł”,
 - Dla sześciu – „GŁÓWNA WYGRANA! Wygrałeś 1 000 000 zł!”

Użytkownik ma kontynuować grę dopóki nie zdecyduje o jej przerwaniu.

5. Napisz program, który implementuje proste szyfry harcerskie: GA-DE-RY-PO-LU-KI, PO-LI-TY-KA-RE-NU, KA-CE-MI-NU-TO-WY. Na początek wyświetl menu, gdzie użytkownik wybiera szyfr. Następnie użytkownik wprowadza ciąg wejściowy, który jest odczytywany przez program. Litery zamieniane są zgodnie ze wzorcem szyfru. Na koniec wyświetl zaszyfrowany tekst.
6. Napisz program, który implementuje szyfr ROT13. Użytkownik wprowadza ciąg wejściowy, który jest odczytywany przez program. Zamień wszystkie małe litery na duże litery. W kolejnym kroku zamień litery zgodnie ze wzorcem szyfru, czyli przesunij każdą z nich o znak znajdujący się 13 liter po nim, np. słowo „HELLO” po zaszyfrowaniu będzie miało postać „URYYB”. Na koniec wyświetl zaszyfrowany tekst.
7. Napisz program, który pobiera od użytkownika łańcuch znaków o maksymalnej długości 40 znaków i sprawdza, czy jest on palindromem. Program powinien zamienić wszystkie duże litery na małe oraz pominąć spacje. Palindromem może być również zdanie, np. „kobyła ma mały bok”.
8. Napisz program kalkulator, który będzie realizował następujące operacje:
 - dodawanie dwóch liczb
 - odejmowanie dwóch liczb
 - dzielenie dwóch liczb
 - mnożenie dwóch liczb
 - wyznaczanie pierwiastka kwadratowego z liczby
 - wyznaczanie procent liczby
 - wyznaczanie reszty z dzielenia dwóch liczb
 - wyznaczanie dowolnej potęgi danej liczby.

Kalkulator powinien umożliwiać wybór operacji tak długo jak tego chce użytkownik.

9. Napisz program służący do konwertowania jednostek temperatury. Użytkownik wprowadza wartość temperatury oraz podaje jedną z trzech możliwych jednostek: Celsjusz, Kelvin lub Fahrenheit. Następnie wybiera z menu, na jaką jednostkę program ma przekonwertować temperaturę. Napisz funkcje konwertujące odpowiednie jednostki i wyświetl wynik na ekranie.
10. Napisz program służący do konwertowania walut. Użytkownik wprowadza kwotę oraz podaje jedną z trzech możliwych walut: złoty, dolar lub euro. Przyjmij, że:
 - 1 USD = 4,38 PLN,
 - 1 EUR = 4,68 PLN.
 Następnie wybiera z menu, na jaką walutę program ma przekonwertować wprowadzoną wartość. Napisz funkcje konwertujące odpowiednie waluty i wyświetl wynik na ekranie.
11. Napisz program do obliczania ilości przełożeń krążków n -piętrowej wieży Hanoi korzystając ze wzoru:
 - rekurencyjnego: $H_n = \begin{cases} 0 & \text{dla } n=0 \\ 2H_{n-1} + 1 & \text{dla } n>0 \end{cases}$
 - zwartego: $H_n = 2^n - 1$

12. Napisz program, który wyświetli trójkąt Pascala od $n=0$ do $n=10$. W tym celu stwórz funkcję obliczającą dwumian Newtona dla podanej liczby n i k całkowitej dodatniej:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}, \quad 0 \leq k \leq n$$

Początkowe wyrazy trójkąta Pascala:

$$\begin{array}{c} \binom{0}{0} \\ \binom{1}{0} \binom{1}{1} \\ \binom{2}{0} \binom{2}{1} \binom{2}{2} \\ \binom{3}{0} \binom{3}{1} \binom{3}{2} \binom{3}{3} \end{array}$$

13. Napisz program, który oblicza BMI (Body Mass Index). Zapytaj użytkownika o płeć, wzrost w cm i wagę w kilogramach. W dalszym etapie zamień centymetry na metry i oblicz BMI korzystając ze

wzoru: $BMI = \text{masa} / \text{wzrost}^2$. Wyświetl odpowiedni komunikat w zależności od wyniku.
Przeprowadź analizę dla 5 osób.

14. Napisz program, który odczytuje liczby od użytkownika do momentu aż użytkownik wprowadzi 0. Zaprojektuj funkcję, która ma zwrócić 0 lub 1 – w zależności od tego czy ilość tych liczb była parzysta czy nieparzysta. Dodatkowo funkcja ma jednocześnie zwrócić sumę oraz iloczyn wszystkich podanych liczb.

15. Napisz program, w którym użytkownik wprowadza ciąg liter. Następnie zamień każdy znak na wartość zgodną z tabelą ASCII. Wynikiem końcowym ma być suma tylko parzystych liczb. Wprowadzone słowo może być pisane z małych lub dużych liter.

Przykład:

INPUT: TeSt

TeSt -> 84, 101, 83, 116

OUTPUT: 84+116 = 200

16. Napisz program obliczający procentową ilość wystąpienia konkretnej cyfry w n rzutach kostką. Najpierw użytkownik określa, dla jakiej liczby będzie dokonywał obliczeń oraz liczbę n rzutów. Następnie zasymuluj n rzutów kostką (cyfry od 1 do 6), wyświetl je i w kolejnym kroku oblicz ile razy dana cyfra wystąpiła w ciągu liczb. Wynik przedstaw w procentach z pominięciem wartości po przecinku.

Przykład:

INPUT: 5 12

OUTPUT: 1 4 2 5 2 1 3 3 5 1 2 2

OUTPUT: 17%

17. Napisz program, w którym użytkownik wprowadza z klawiatury dowolny znak. Wyświetl na ekranie znak odpowiadający połowie wartości wprowadzonego znaku uzyskanego za pomocą tabeli ASCII. Zaokrąglij wynik dzielenia w dół. Dla przykładu, gdy użytkownik wprowadzi znak Q, to wynikiem działania programu powinien być znak (. Program powinien działać tak długo aż użytkownik wpisze znak 0.

18. Napisz program do obliczania wyniku funkcji Ackermanna korzystając ze wzoru:

$$A(m, n) = \begin{cases} n + 1 & \text{dla } m=0 \\ A(m-1, 1) & \text{dla } m>0 \text{ i } n=0 \\ A(m-1, A(m, n-1)) & \text{dla } m>0 \text{ i } n>0 \end{cases}$$

Zastosuj w programie ograniczenia: $0 \leq n \leq 4$, $0 \leq m \leq 3$.

19. Napisz program sprawdzający, czy wprowadzona przez użytkownika liczba naturalna jest liczbą szczęśliwą. Liczba szczęśliwa to taka, dla której obliczamy sumę kwadratów jej cyfr, tak długo aż dostaniemy 1. Dla przykładu liczbą szczęśliwą jest 28, ponieważ:

- krok 1: $2^2 + 8^2 = 68$
- krok 2: $6^2 + 8^2 = 100$
- krok 3: $1^2 + 0^2 + 0^2 = 1$

Liczba 4 nie jest liczbą szczęśliwą, ponieważ: 4 -> 16 -> 37 -> 58 -> 89 -> 145 -> 42 -> 20 -> 4.

Wyświetl na ekranie poszczególne kroki oraz komunikat, czy dana liczba jest szczęśliwa.

20. Napisz program, który wprowadza ciąg liczb naturalnych oddzielonych spacjami i wyświetla znaki odpowiadające tym wartościom w tabeli ASCII. Jeśli znaku nie można wyświetlić, ma on być przedstawiony w systemie szesnastkowym poprzedzony ukośnikiem. Przykład:

INPUT: 72 101 76 108 79

OUTPUT: HeLlO

INPUT: 31 125 68 123 150

OUTPUT: \1F}D{\96