

### Zadania na ćwiczenia 1

1. Napisać program wypisujący elementy ciągu Fibonacciego mniejsze od miliona.
2. Znaleźć wyrazy początkowe zamiast 1,1 o najmniejszej sumie, aby w ciągu analogicznym do ciągu Fibonacciego wystąpił wyraz równy numerowi bieżącego roku.
3. Napisać program sprawdzający czy istnieje spójny podciąg ciągu Fibonacciego o zadanej sumie.
4. Napisać program obliczający pierwiastek całkowitoliczbowy z liczby naturalnej korzystając z zależności  $1+3+5+\dots = n^2$ .
5. Napisać program wyznaczający pierwiastek kwadratowy ze wzoru Newtona.
6. Napisać program rozwiązujący równanie  $x^x=2019$  metodą bisekcji.

### Zadania dodatkowe

7. Napisz program wczytujący liczbę naturalną z klawiatury i odpowiadający na pytanie, czy liczba ta jest iloczynem dowolnych dwóch kolejnych wyrazów ciągu Fibonacciego.
8. Napisać program sprawdzający czy zadana liczba jest pierwsza.
9. Napisać program wypisujący dzielniki liczby.
10. Napisać program wyszukujący liczby doskonałe mniejsze od miliona.
11. Napisać program wyszukujący liczby zaprzyjaźnione mniejsze od miliona.
12. Napisać program wyznaczający największy wspólny dzielnik 3 zadanych liczb.
13. Napisać program wyznaczający najmniejszą wspólną wielokrotność 3 zadanych liczb.
14. Napisać program obliczający wartości  $\cos(x)$  z rozwinięcia w szereg Maclaurina.
15. Nieskończony iloczyn  $\sqrt{0.5} * \sqrt{0.5 + 0.5 * \sqrt{0.5}} * \sqrt{0.5 + 0.5 * \sqrt{0.5 + 0.5 * \sqrt{0.5}}} * \dots$  ma wartość  $2/\pi$ . Napisz program korzystający z tej zależności i wyznaczający wartość  $\pi$ .
16. Dany jest ciąg określony wzorem:  $A(n+1) = (A(n)\%2) * (3*A(n)+1) + (1-A(n)\%2) * A(n) / 2$  Startując z dowolnej liczby naturalnej  $>1$  ciąg ten osiąga wartość 1. Napisać program, który znajdzie wyraz początkowy z przedziału 2-10000 dla którego wartość 1 jest osiągalna po największej liczbie kroków.
17. Napisać program wyznaczający wartość do której zmierza iloraz dwóch kolejnych wyrazów ciągu Fibonacciego. Wyznaczyć ten iloraz dla różnych wartości początkowych wyrazów ciągu.
18. Zmodyfikować wzór Newtona aby program z zadania 7 obliczał pierwiastek stopnia 3.
19. Napisać program wyznaczający wartość liczby  $e$  korzystając z zależności:  $e = 1/0! + 1/1! + 1/2! + 1/3! + \dots$
20. Dane są ciągi:  $A(n+1)=\sqrt{A(n)*B(n)}$  oraz  $B(n+1)=(A(n)+B(n))/2.0$ . Ciągi te są zbieżne do wspólnej granicy nazywanej średnią arytmetyczno-geometryczną. Napisać program wyznaczający średnią arytmetyczno-geometryczną dwóch liczb.

## Zadania na ćwiczenia 2

1. Napisz program wczytujący liczbę naturalną z klawiatury i odpowiadający na pytanie, czy liczba ta jest iloczynem dowolnych dwóch wyrazów ciągu Fibonacciego.
2. Napisz program wczytujący liczbę naturalną z klawiatury i odpowiadający na pytanie, czy liczba ta jest wielokrotnością dowolnego wyrazu ciągu danego wzorem  $A(n)=n*n+n+1$ .
3. Napisz program wczytujący liczbę naturalną z klawiatury i odpowiadający na pytanie, czy liczba naturalna jest palindromem, a następnie czy jest palindromem w systemie dwójkowym.
4. Liczba dwu-trzy-piątkowa w rozkładzie na czynniki pierwsze nie posiada innych czynników niż 2,3,5. Jedynek też jest taką liczbą. Napisz program, który wylicza ile takich liczb znajduje się w przedziale od 1 do N włącznie.
5. Napisać program, który wyznacza ostatnią niezerową cyfrę  $N!$ . Program powinien działać dla  $N$  rzędu 1000000.
6. Dana jest liczba naturalna o niepowtarzających się cyfrach spośród których nie ma zera. Ile różnych liczb podzielnych np. przez 7 można otrzymać poprzez wykreślenie dowolnych cyfr w tej liczbie. Np. dla 2315 będą to 21, 35, 231, 315.
7. Napisać program wczytujący liczbę naturalną z klawiatury i rozkładający ją na iloczyn 2 liczb o najmniejszej różnicy. Np.  $30=5*6$ ,  $120=10*12$ .
8. Napisać program wczytujący trzy liczby naturalne  $a, b, n$  i wypisujący rozwinięcie dziesiętne ułamka  $a/b$  z dokładnością do  $n$  miejsc po kropce dziesiętnej. ( $n$  jest rzędu 100)
9. Pewnych liczb nie można przedstawić jako sumy elementów spójnych fragmentów ciągu Fibonacciego, np. 9,14,15,17,22. Proszę napisać program, który wczytuje liczbę naturalną  $n$ , wylicza i wypisuje następną taką liczbę większą od  $n$ . Można założyć, że  $0 < n < 1000$ .
10. Napisać program, który oblicza pole figury pod wykresem funkcji  $y=1/x$  w przedziale od 1 do  $k$ , metodą prostokątów.

## Zadania dodatkowe

11. Napisz program wczytujący liczbę naturalną z klawiatury i odpowiadający na pytanie, czy liczba ta jest wielokrotnością dowolnego wyrazu ciągu danego wzorem  $A(n)=3*A(n-1)+1$ , a pierwszy wyraz jest równy 2.
12. Napisz program wczytujący liczbę naturalną z klawiatury i odpowiadający na pytanie, czy jej cyfry stanowią ciąg rosnący.
13. Proszę napisać program, który wczytuje wprowadzany z klawiatury ciąg liczb naturalnych zakończonych zerem stanowiącym wyłącznie znacznik końca danych. Program powinien wypisać te elementy ciągu które są równe średniej arytmetycznej z 4 najbliższych sąsiadów. Na przykład dla ciągu: 2,3,2,7,1,2,4,8,5,2,2,4,3,9,5,4,0 powinny zostać wypisane podkreślone liczby. Można założyć, że w ciągu znajduje się co najmniej 5 elementów.
14. Napisz program wczytujący liczbę naturalną z klawiatury i odpowiadający na pytanie, czy liczba ta zawiera cyfrę równą liczbie swoich cyfr.
15. Napisz program wczytujący liczbę naturalną z klawiatury i odpowiadający na pytanie, czy liczba zakończona jest unikalną cyfrą.
16. Dane są dwie liczby naturalne z których budujemy trzecią liczbę. W budowanej liczbie muszą wystąpić wszystkie cyfry występujące w liczbach wejściowych. Wzajemna kolejność cyfr każdej z liczb wejściowych musi być zachowana. Na przykład mając liczby 123 i 75 możemy zbudować liczby 12375, 17523, 75123, 17253, itd. Proszę napisać funkcję która wyznaczy ile liczb pierwszych można zbudować z dwóch zadanych

liczb.

17. Napisać program znajdujący wszystkie liczby N-cyfrowe dla których suma N-tych potęg cyfr liczby jest równa tej liczbie, np.  $153=1^3+5^3+3^3$ .
18. Liczba Smitha to taka, której suma cyfr jest równa sumie cyfr wszystkich liczb występujących w jej rozkładzie na czynniki pierwsze. Na przykład:  $85=5*17$ ,  $8+5=5+1+7$ . Napisać program wypisujący liczby Smitha mniejsze od 1000000.
19. Napisać program poszukujący trójkątów Pitagorejskich.
20. Napisać program wyliczający pierwiastek równania  $x^x=2017$  metodą stycznych.
21. Mamy dane dwa ciągi o następujących zależnościach:  
A:  $a_0 = 0, a_1 = 1, a_n = a_{n-1} - b_{n-1} * a_{n-2}$   
B:  $b_0 = 2, b_n = b_{n-1} + 2 * a_{n-1}$   
Proszę napisać program, który czyta liczby typu *int* ze standardowego wejścia i tak długo jak liczby te są kolejnymi wyrazami ciągu  $A_n$  (tj.  $a_0, a_1, a_2, \dots$ ) wypisuje na standardowe wyjście wyrazy drugiego ciągu  $B_n$  (tj.  $b_0, b_1, b_2, \dots$ ).
22. Napisać program wczytujący dwie liczby naturalne  $a, b$  i wypisujący rozwinięcie dziesiętne ułamka  $a/b$  w postaci ułamka okresowego. Na przykład  $1/3 = 0.(3)$ ,  $1/6 = 0.1(6)$ ,  $1/7 = 0.(142857)$
23. Dwie liczby naturalne są różno-cyfrowe jeżeli nie posiadają żadnej wspólnej cyfry. Proszę napisać program, który wczytuje dwie liczby naturalne i poszukuje najmniejszej podstawy systemu (w zakresie 2-16) w którym liczby są różno-cyfrowe. Program powinien wypisać znalezioną podstawę, jeżeli podstawa taka nie istnieje należy wypisać komunikat o jej braku. Na przykład: dla liczb 123 i 522 odpowiedzią jest podstawa 11 bo  $123_{(10)}=102_{(11)}$  i  $522_{(10)}=435_{(11)}$ .

### **Zadania na ćwiczenia 3**

Uwaga: w poniższych zadaniach MAX jest zdefiniowaną stałą.

1. Napisać program zamieniający liczbę naturalną z systemu 10 na podstawę 2-16
2. Napisać program wczytujący dwie liczby naturalne i odpowiadający na pytanie czy są one zbudowane z takich samych cyfr, np. 123 i 321, 1255 i 5125, 11000 i 10001.
3. Napisać program generujący i wypisujący liczby pierwsze mniejsze od N metodą Sita Eratostenesa
4. Napisać program obliczający i wypisujący wartość  $N!$  dla N z zakresu od 1 do 1000.
5. Obliczanie stałej e z rozwinięcia w szereg  $e=1/0!+1/1!+1/2!+1/3!+\dots$  z dokładnością do np. 1000 cyfr dziesiętnych.
6. Proszę napisać program, który wczytuje wprowadzany z klawiatury ciąg liczb naturalnych zakończonych zerem stanowiącym wyłącznie znacznik końca danych. Program powinien wypisać 10 co do wielkości wartość, jaka wystąpiła w ciągu. Można założyć, że w ciągu znajduje się wystarczająca liczba elementów.

### **Zadania dodatkowe**

7. Napisać program wypełniający tablicę `int t[MAX]` liczbami naturalnymi 1-1000 i sprawdzający czy każdy element tablicy zawiera co najmniej jedną cyfrę nieparzystą.
8. Napisać program wypełniający tablicę `int t[MAX]` liczbami naturalnymi 1-1000 i sprawdzający czy istnieje element tablicy zawierający wyłącznie cyfry nieparzyste.
9. Dana jest tablica `int t[N]` zawierająca liczby naturalne. W tablicy możemy przeskoczyć z pola o indeksie k o n pól w prawo jeżeli wartość n jest czynnikiem pierwszym liczby `t[k]`. Napisać funkcję sprawdzającą czy jest możliwe przejście z pola 0 na pole N-1.
10. Napisać funkcję, która dla tablicy `int t[MAX]` wypełnionej liczbami naturalnym wyznacza długość najdłuższego, spójnego podciągu rosnącego.
11. Napisać funkcję, która dla tablicy `int t[MAX]` wypełnionej liczbami naturalnym wyznacza długość najdłuższego, spójnego podciągu arytmetycznego.

12. Napisać funkcję, która dla tablicy `int t[MAX]` wypełnionej liczbami naturalnym wyznacza długość najdłuższego, spójnego podciągu geometrycznego.
13. Proszę napisać program, który wypełnia tablicę `t[N]` pseudolosowymi liczbami nieparzystymi z zakresu `[1..99]`, a następnie Wyznacza i wypisuje różnicę pomiędzy długością najdłuższego znajdującego się w niej ciągu arytmetycznego o dodatniej różnicy, a długością najdłuższego ciągu arytmetycznego o ujemnej różnicy, przy założeniu, że kolejnymi wyrazami ciągu są elementy tablicy o kolejnych indeksach.
14. Proszę napisać program, który wypełnia tablice `int tab[MAX]` trzycyfrowymi liczbami pseudolosowymi, a następnie wyznacza i wypisuje długość najdłuższego podciągu spójnego znajdującego się w tablicy dla którego w tablicy występuje również rewers tego ciągu. Na przykład dla tablicy: 2,9,3,1,7,11,9,6,7,7,1,3,9,12,15 odpowiedzią jest liczba 4.
15. Napisać program wyznaczający na drodze eksperymentu prawdopodobieństwo tego, że w grupie `N` przypadkowo spotkanych osób, co najmniej dwie urodziły się tego samego dnia roku. Wyznaczyć wartości prawdopodobieństwa dla `N` z zakresu 20-40.
16. Dana jest duża tablica `int t[MAX]`. Proszę napisać funkcję, która zwraca informację czy w tablicy zachodzi następujący warunek: „wszystkie elementy, których indeks jest elementem ciągu Fibonacciego są liczbami złożonymi, a wśród pozostałych przynajmniej jedna jest liczbą pierwszą”
17. Mamy zdefiniowaną `n`-elementową tablicę liczb całkowitych. Proszę napisać funkcję zwracającą wartość typu `bool` oznaczającą, czy w tablicy istnieje dokładnie jeden element najmniejszy i dokładnie jeden element największy (liczba elementów najmniejszych oznacza liczbę takich elementów o tej samej wartości).
18. Funkcja wyznacza różnicę pomiędzy wartością maksymalną a minimalną. Ile minimalnie porównań trzeba wykonać?
19. Napisać funkcję, do której przekazujemy licznik i mianownik ułamka. Funkcja powinna wypisać wartość ułamka jako rozwinięcie dziesiętne uwzględniające okres, np.  $1/3=0.(3)$ ,  $1/6=0.1(6)$ .
20. Dane są dwie tablice `t1[N]` i `t2[N]` zawierające liczby naturalne. Z wartości w obu tablicach możemy tworzyć sumy. „Poprawna” suma to taka, która zawiera co najmniej jeden element (z tablicy `t1` lub `t2`) o każdym indeksie. Na przykład dla tablic: `t1 = [1,3,2,4]` i `t2 = [9,7,4,8]` poprawnymi sumami są na przykład  $1+3+2+4$ ,  $9+7+4+8$ ,  $1+7+3+8$ ,  $1+9+7+2+4+8$ . Proszę napisać funkcje generującą i wypisującą wszystkie poprawne sumy, które są

liczbami pierwszymi. Do funkcji należy przekazać dwie tablice, funkcja powinna zwrócić liczbę znalezionych i wypisanych sum.

21. Dane są deklaracje:

```
const int N=1000;  
int tab[N];
```

Tablica `tab` jest wypełniona liczbami naturalnymi. Proszę napisać funkcję, która zwraca długość najdłuższego spójnego podciągu będącego palindromem złożonym wyłącznie z liczb nieparzystych. Do funkcji należy przekazać tablicę, funkcja powinna zwrócić długość znalezionego podciągu lub wartość 0 jeżeli taki podciąg nie istnieje.

22. Dane są deklaracje:

```
const int N=1000;  
int tab[N];
```

Tablica `tab` jest wypełniona liczbami naturalnymi. Proszę napisać funkcję, która zwraca długość najdłuższego, spójnego podciągu rosnącego dla którego suma jego elementów jest równa sumie indeksów tych elementów. Do funkcji należy przekazać tablicę, funkcja powinna zwrócić długość znalezionego podciągu lub wartość 0 jeżeli taki podciąg nie istnieje.

23. Dana jest tablica `int t[N]` zawierająca liczby naturalne mniejsze od 1000. Proszę napisać funkcję, która zwraca długość najdłuższego, spójnego fragmentu tablicy, dla którego w iloczynie jego elementów każdy czynnik pierwszy występuje co najwyżej raz. Na przykład dla tablicy `[2,23,33,35,7,4,6,7,5,11,13,22]` wynikiem jest wartość 5.

24.

#### Zadania na ćwiczenia 4

Uwaga: w poniższych zadaniach MAX jest zdefiniowaną stałą.

1. Dana jest tablica `int t[MAX][MAX]`. Proszę napisać funkcję wypełniającą tablicę liczbami naturalnymi po spirali.
2. Dana jest tablica `int t[MAX][MAX]` wypełniona liczbami naturalnymi. Proszę napisać funkcję która odpowiada na pytanie, czy w każdym wierszu tablicy występuje co najmniej jedna liczba złożona wyłącznie z nieparzystych cyfr.
3. Dana jest tablica `int t[MAX][MAX]` wypełniona liczbami naturalnymi. Proszę napisać funkcję która odpowiada na pytanie, czy istnieje wiersz w tablicy w którym każda z liczb zawiera przynajmniej jedną cyfrę parzystą.
4. Dana jest tablica `int t[MAX][MAX]` wypełniona liczbami naturalnymi. Proszę napisać funkcję która zwraca wiersz i kolumnę dowolnego elementu, dla którego iloraz sumy elementów w kolumnie w którym leży element do sumy elementów wiersza w którym leży element jest największa.
5. Poprzednie zadanie z tablicą wypełnioną liczbami całkowitymi.

#### Zadania dodatkowe

6. Dane są dwie tablice mogące pomieścić taką samą liczbę elementów:  

```
int t1[MAX][MAX];  
int t2[MAX2];           // MAX2 = MAX*MAX
```

W każdym wierszu tablicy t1 znajdują się uporządkowane rosnąco (w obrębie wiersza) liczby naturalne. Proszę napisać funkcję przepisującą wszystkie singletony (liczby występujące dokładnie raz) z tablicy t1 do t2, tak aby liczby w tablicy t2 były uporządkowane rosnąco. Pozostałe elementy tablicy t2 powinny zawierać zera.
7. Dane są dwie tablice mogące pomieścić taką samą liczbę elementów:  

```
int t1[MAX][MAX];  
int t2[MAX2];           // MAX2 = MAX*MAX
```

W każdym wierszu tablicy t1 znajdują się uporządkowane niemalejąco (w obrębie wiersza) liczby naturalne. Proszę napisać funkcję przepisującą wszystkie liczby z tablicy t1 do t2, tak aby liczby w tablicy t2 były uporządkowane niemalejąco.
8. Dana jest tablica wypełniona liczbami naturalnymi:  

```
int t[MAX][MAX];
```

Proszę napisać funkcję, która w poszukiwuje w tablicy najdłuższego ciągu geometrycznego leżącego ukośnie w kierunku prawo-dół, liczącego co najmniej 3 elementy. Do funkcji należy przekazać tablicę. Funkcja powinna zwrócić informacje czy udało się znaleźć taki ciąg oraz długość tego ciągu.
9. Dana jest tablica wypełniona liczbami naturalnymi:  

```
int t[MAX][MAX];
```

Proszę napisać funkcję, która w poszukiwuje w tablicy kwadratu o liczbie pól będącej liczbą nieparzystą większą od 1, którego iloczyn 4 pól narożnych wynosi k. Do funkcji należy przekazać tablicę i wartość k. Funkcja powinna zwrócić informacje czy udało się znaleźć kwadrat oraz współrzędne (wiersz, kolumna) środka kwadratu.
10. Napisać funkcję która dla tablicy typu `int t[MAX][MAX]`, wypełnionej liczbami całkowitymi, zwraca wartość `true` w przypadku, gdy w każdym wierszu i każdej

kolumnie występuje co najmniej jedno 0 oraz wartość false w przeciwnym przypadku.

11. Dwie liczby naturalne są „przyjaciółkami” jeżeli zbiory cyfr z których zbudowane są liczby są identyczne. Na przykład: 123 i 321, 211 i 122, 35 3553. Dana jest tablica `int t[MAX][MAX]` wypełniona liczbami naturalnymi. Proszę napisać funkcję, która dla tablicy `t` zwraca ile elementów tablicy sąsiaduje wyłącznie z przyjaciółkami
12. Dana jest tablica `int t[MAX][MAX][MAX]`. Proszę napisać funkcję, do której przekazujemy tablicę wypełnioną liczbami większymi od zera. Funkcja powinna zwracać wartość `True`, jeżeli na wszystkich poziomach tablicy liczba elementów sąsiadujących (w obrębia poziomu) z co najmniej 6 liczbami złożonymi jest jednakowa albo wartość `False` w przeciwnym przypadku.
13. Liczby naturalne `a, b` są komplementarne jeżeli ich suma jest liczbą pierwszą. Dana jest tablica typu `int t[MAX][MAX]` wypełniona liczbami naturalnymi. Proszę napisać funkcję, która zeruje elementy nie posiadające liczby komplementarnej.
14. Dwie liczby naturalne są zgodne jeżeli w zapisie dwójkowym zawierają tę samą liczbę jedynek, np.  $22 = 10110_2$  i  $14 = 1110_2$ . Dane są tablice ( $MAX2 > MAX1$ ):  
`int tab1[MAX1][MAX1];`  
`int tab2[MAX2][MAX2];`  
Proszę napisać funkcję, która sprawdza czy istnieje takie położenie tablicy typu `tab1` wewnątrz tablicy typu `tab2`, przy którym liczba zgodnych elementów jest większa od 33%. Do funkcji należy przekazać tablicę typu `tab1` i tablicę typu `tab2`. Obie oryginalne tablice powinny pozostać nie zmieniane.
15. Dana jest tablica `int t[MAX][MAX]`, wypełniona liczbami naturalnymi. Proszę napisać funkcję, która odpowiada na pytanie, czy w tablicy istnieje wiersz, w którym każda liczba zawiera co najmniej jedną cyfrę będącą liczbą pierwszą?
16. Dana jest tablica `int t[MAX][MAX]`, wypełniona liczbami naturalnymi. Proszę napisać funkcję która odpowiada na pytanie, czy w tablicy każdy wiersz zawiera co najmniej jedną liczbą złożoną wyłącznie z cyfr będących liczbami pierwszymi?
17. Dana jest tablica `int t[MAX][MAX]` wypełniona liczbami naturalnymi. Proszę napisać funkcję która zwraca wiersz i kolumnę dowolnego elementu, dla którego suma otaczających go elementów jest największa.
18. Dana jest tablica typu `int t[MAX][MAX]` wypełniona liczbami całkowitymi. Proszę napisać **funkcję**, która wyszuka spójny podciąg elementów leżący poziomo lub pionowo o największej sumie. Maksymalna długość podciagu może wynosić 10 elementów. Do funkcji należy przekazać tablicę typu `tab`, funkcja powinna zwrócić sumę maksymalnego podciagu.
19. Dana jest tablica wypełniona liczbami naturalnymi `int t[MAX][MAX];`  
Proszę napisać funkcję, która zwraca liczbę par elementów, o określonym iloczynie, takich że elementy są odległe o jeden ruch skoczka szachowego.
20. Dana jest tablica `t[N][N]` (reprezentująca szachownicę) wypełniona liczbami naturalnymi. Proszę napisać funkcję która ustawia na szachownicy dwie wieże, tak aby suma liczb na „szachowanych” przez wieże polach była największa. Do funkcji należy przekazać tablicę, funkcja powinna zwrócić położenie wież. Uwaga- zakładamy, że wieża szachuje cały wiersz i kolumnę z wyłączeniem pola na którym stoi



21. Dana jest tablica `int t[N][N]` (reprezentująca szachownicę) wypełniona liczbami naturalnymi. W każdej kolumnie znajduje się dokładnie jedna wieża, której numer wiersza zawiera tablica `int w[N]`. Proszę napisać funkcję która wybiera do usunięcia z szachownicy dwie wieże, tak aby suma liczb na polach szachowanych przez pozostałe wieże była najmniejsza. Do funkcji należy przekazać tablice `t` i `w`, funkcja powinna zwrócić numery kolumn z których usunięto wieże. Uwaga - zakładamy, że wieża szachuje cały wiersz i kolumnę z wyłączeniem pola na którym stoi
22. Dana jest tablica `int t[N][N]` wypełniona przypadkowymi wartościami. Proszę napisać funkcję, która dla zmiennej typu tablica zwraca numer wiersza w którym występuje najdłuższy spójny fragment złożony z liczb o tej samej wartości. W przypadku kilku fragmentów o tej samej długości należy zwrócić pozycję pierwszego z nich.

## Zadania na ćwiczenia 5

1. Liczby wymierne są reprezentowane przez strukturę:

```
struct ulamek {
    int l; // licznik - liczba całkowita
    int m; // mianownik - liczba większa od 0
};
```

Proszę napisać podstawowe operacje na ułamkach, m.in. dodawanie, odejmowanie, mnożenie, dzielenie, potęgowanie, skracanie, wypisywanie i wczytywanie.

2. Używając funkcji z poprzedniego zadania proszę napisać funkcję rozwiązującą układ 2 równań o 2 niewiadomych.

3. Na szachownicy o wymiarach 100x100 umieszczamy N hetmanów ( $N < 100$ ). Położenie hetmanów jest opisywane przez strukturę dane:

```
struct hetman {
    int w; // wiersz 0..99
    int k; // kolumna 0..99
};
```

```
struct dane {
    hetman t[100]; // położenie hetmanów
    int N;          // liczba hetmanów
};
```

Proszę napisać funkcję, która odpowiada na pytanie: czy żadne z dwa hetmany się nie szachują? Do funkcji należy przekazać położenie hetmanów.

4. Dana jest tablica ulamek `t[100]` zawierająca ciąg liczb wymiernych. Proszę napisać funkcję, która policzy występujące w tablicy ciągi arytmetyczne (LA) i geometryczne (LG) o długości większej niż 2. Funkcja powinna zwrócić wartość 1 gdy  $LA > LG$ , wartość -1 gdy  $LA < LG$  oraz 0 gdy  $LA = LG$ .

5. Dany jest zbiór punktów leżących na płaszczyźnie opisany przy pomocy struktur:

```
struct punkt {
    int x; // zakres wsp. 0..31
    int y; // zakres wsp. 0..31
};

struct dane {
    punkt t[100]; // położenia punktów
    int N;        // liczba punktów
};
```

Proszę napisać funkcję, która zwraca wartość `true` jeżeli zbiorze istnieją 4 punkty wyznaczające kwadrat o bokach równoległych do osi układu współrzędnych, a wewnątrz tego kwadratu nie ma żadnych innych punktów. Do funkcji należy przekazać strukturę opisującą położenie punktów.

## Zadania dodatkowe

6. Liczby zespolone są reprezentowane przez strukturę:

```
struct ulamek {
    double re;
    double im;
};
```

Proszę napisać podstawowe operacje na liczbach zespolonych, m.in. dodawanie, odejmowanie, mnożenie, dzielenie, potęgowanie, wypisywanie i wczytywanie.

7. Używając funkcji z poprzedniego zadania proszę napisać funkcję rozwiązującą równanie kwadratowe o współczynnikach zespolonych.

### **Zadania na ćwiczenia 6**

1. Dany jest zestaw odważników `int t[N]`. Napisać funkcję, która sprawdza czy jest możliwe odważenie określonej masy. Odważniki można umieszczać tylko na jednej szalce.
2. Poprzednie zadanie, odważniki można umieszczać na obu szalkach.
3. Poprzednie zadanie. Program powinien wypisywać wybrane odważniki.
4. „Waga” liczby jest określona jako ilość różnych czynników pierwszych liczby. Na przykład `waga(1)=0`, `waga(2)=1`, `waga(6)=2`, `waga(30)=3`, `waga(64)=1`. Dana jest tablica: `int t[N]`. Tablica zawiera liczby naturalne. Proszę napisać funkcję, która sprawdza czy można elementy tablicy podzielić na 3 podzbiory o równych wagach. Do funkcji należy przekazać wyłącznie tablicę, funkcja powinna zwrócić wartość typu `bool`.

### **Zadania na ćwiczenia 7**

5. Problem skoczka szachowego (treść oczywista)
6. Dany jest ciąg zer i jedynek zapisany w tablicy `int t[N]`. Proszę napisać funkcję, która odpowiada na pytanie czy jest możliwe pocięcie ciągu na kawałki z których każdy reprezentuje liczbę pierwszą. Długość każdego z kawałków nie może przekraczać 30. Na przykład dla ciągu 111011 jest to możliwe, a dla ciągu 110100 nie jest możliwe.
7. Dana jest tablica `int t[N]`. Proszę napisać funkcję, która znajdzie niepusty, najmniejszy (w sensie liczebności) podzbiór elementów tablicy, dla którego suma elementów jest równa sumie indeksów tych elementów. Do funkcji należy przekazać tablicę, funkcja powinna zwrócić sumę elementów znalezionej podzbioru. Na przykład dla tablicy: [ 1, 7, 3, 5, 11, 2 ] rozwiązaniem jest liczba 10.
8. Szachownica jest reprezentowana przez tablicę `int t[8][8]` wypełnioną liczbami naturalnymi zawierającymi koszt przebywania na danym polu szachownicy. Król szachowy znajduje się w wierszu 0 i kolumnie k. Król musi w dokładnie 7 ruchach dotrzeć do wiersza 7. Proszę napisać funkcję, która wyznaczy minimalny koszt przejścia króla. Do funkcji należy przekazać tablicę t oraz startową kolumnę k. Koszt przebywania na polu startowym i ostatnim także wliczamy do kosztu przejścia.

### **Zadania dodatkowe**

9. Rekurencyjne obliczanie wyznacznika z macierzy (treść oczywista)
10. Dana jest tablica `int t[N]`. Proszę napisać program zliczający liczbę „enek” o określonym iloczynie.
11. Proszę zmodyfikować poprzedni program aby wypisywał znalezione n-ki.
12. Napisać program wypisujący wszystkie możliwe podziały liczby naturalnej na sumę składników. Na przykład dla liczby 4 są to: 1+3, 1+1+2, 1+1+1+1, 2+2.
13. Problem wież w Hanoi (treść oczywista)
14. Problem 8 Hetmanów (treść oczywista)
15. Wyrazy budowane są z liter a..z. Dwa wyrazy „ważą” tyle samo jeżeli: mają tę samą liczbę samogłosek oraz sumy kodów ascii liter z których są zbudowane są identyczne, na przykład „ula” -> 117 108 97 oraz „exe” 101 120 101. Proszę napisać funkcję `bool wyraz( string s1, string s2)`, która sprawdza czy jest możliwe zbudowanie wyrazu z podzbioru liter zawartych w s2 ważącego tyle co wyraz s1. Dodatkowo funkcja powinna wypisać znaleziony wyraz.

16. Dane są dwie liczby naturalne z których budujemy trzecią liczbę. W budowanej liczbie muszą wystąpić wszystkie cyfry występujące w liczbach wejściowych. Wzajemna kolejność cyfr każdej z liczb wejściowych musi być zachowana. Na przykład mając liczby 123 i 75 możemy zbudować liczby 12375, 17523, 75123, 17253, itd. Proszę napisać funkcję która wyznaczy ile liczb pierwszych można zbudować z dwóch zadanych liczb.
17. W szachownicy o wymiarach 8x8 każdemu z pól przypisano liczbę naturalną. Na ruchy króla nałożono dwa ograniczenia: król może przesunąć się na jedno z 8 sąsiednich pól jeżeli ostatnia cyfra liczby na polu na którym stoi jest mniejsza od pierwszej cyfry liczby pola docelowego, oraz w drodze do obranego celu (np. narożnika) król nie może wykonać ruchu, który powoduje oddalenie go od celu. Dana jest globalna tablica `int t[8][8]` wypełniona liczbami naturalnymi reprezentująca szachownicę. Lewy górny narożnik ma współrzędne `w=0` i `k=0`. Proszę napisać funkcję sprawdzającą czy król może dostać się z pola `w,k` do prawego dolnego narożnika .
18. Zadanie jak powyżej. Funkcja sprawdzająca czy król może dostać się z pola `w,k` do któregośkolwiek z narożników.
19. Zadanie jak powyżej. Funkcja powinna dostarczyć drogę króla w postaci tablicy zawierającej kierunki (liczby od 0 do 7) poszczególnych ruchów króla do wybranego celu.
20. Tablica `int t[8][8]` zawiera liczby naturalne. Proszę napisać funkcję, która sprawdza czy można wybrać z tablicy niepusty podzbiór o zadanej sumie. Warunkiem dodatkowym jest aby żadne dwa wybrane elementy nie leżały w tej samej kolumnie ani wierszu. Do funkcji należy przekazać wyłącznie tablicę oraz wartość sumy, funkcja powinna zwrócić wartość typu `bool`.
21. Dana jest tablica `int t[N]` zawierająca liczby naturalne. Po tablicy możemy przemieszczać się według następującej zasady: z pola o indeksie `i` możemy przeskoczyć na pole o indeksie `i+k` jeżeli `k` jest czynnikiem pierwszym liczby `t[i]` mniejszym od `t[i]`. Proszę napisać funkcję, która zwraca informację czy jest możliwe przejście z pola o indeksie 0 na pole o indeksie `N-1`. Funkcja powinna zwrócić liczbę wykonanych skoków lub wartość `-1` jeżeli powyższe przejście nie jest możliwe.
22. Dana jest tablica `int t[N]` zawierająca oporności `N` rezystorów wyrażonych całkowitą liczbą `kΩ`. Proszę napisać funkcję, która sprawdza czy jest możliwe uzyskanie wypadkowej rezystancji `R` (równej całkowitej liczbie `kΩ`) łącząc dowolnie 3 wybrane rezystory.
23. Dane są definicje:
- ```
struct punkt { float x,y; };
punkt t[N];
```
- Tablica `t` zawiera położenia `N` punktów. Napisz funkcję, która zwróci najmniejszą odległość między środkami ciężkości 2 niepustych podzbiorów tego zbioru.
24. Napisz procedurę, która jako argument przyjmuje liczbę całkowitą `i` wypisuje wszystkie co najmniej dwucyfrowe liczby pierwsze, powstałe poprzez wykreślenie z liczby pierwotnej co najmniej jednej cyfry.
25. Tablica `t[N]` jest wypełniona liczbami naturalnymi. Skok z pola `i`-tego można wykonać na pola o indeksach `i+k`, gdzie `k` jest czynnikiem pierwszym liczby `t[i]` (mniejszym od niej samej). Napisz funkcję, która sprawdza, czy da się przejść z pola 0 do `N-1` – jeśli się da, zwraca ilość skoków, jeśli się nie da, zwraca `-1`.
26. Do budowy liczby naturalnej reprezentowanej w systemie dwójkowym możemy użyć `A` cyfr 1 oraz `B` cyfr 0, gdzie `A,B>0`. Proszę napisać funkcję, która dla zadanych parametrów `A` i `B` zwraca ilość wszystkich możliwych do zbudowania liczb, takich że pierwsza cyfra w systemie dwójkowym (najstarszy bit) jest równa 1, a zbudowana liczba jest

złożona. Na przykład dla  $A=2$ ,  $B=3$  ilość liczb wynosi 3, są to 10010(2) 10100(2) 11000(2)

27. Dane są definicje:

```
struct kwadrat { int x1,x2,y1,y2; };  
kwadrat t[N];
```

gdzie  $x_1, x_2, y_1, y_2$  oznaczają proste ograniczające kwadrat ( $x_1 < x_2, y_1 < y_2$ ). Proszę napisać funkcję, która zwraca wartość logiczną true, jeśli danej tablicy można znaleźć 13 nienachodzących na siebie kwadratów, których suma pól jest rów na 2012 i false w przeciwnym przypadku.

28. Dany jest zbiór  $n$  liczb naturalnych umieszczony w tablicy `int t[N]`. Proszę napisać funkcję, która zwraca informację, czy jest możliwy podział zbioru  $n$  liczb na trzy podzbiory, tak aby w każdym podzbiorze, łączna liczba jedynek użyta do zapisu elementów tego podzbioru w systemie dwójkowym była jednakowa. Na przykład: {2,3,5,7,15} -> true, bo podzbiory {2,7} {3,5} {15} wymagają użycia 4 jedynek, {5,7,15} -> false, podział nie istnieje.

29. Dana jest tablica `int t[N]`. Proszę napisać funkcję, która znajdzie najmniejszy (w sensie liczebności) podzbiór elementów tablicy, dla którego suma elementów jest równa sumie indeksów tych elementów. Do funkcji należy przekazać tablicę, funkcja powinna zwrócić sumę elementów znalezionego podzbioru. Na przykład dla tablicy: [ 1, 7, 3, 5, 11, 2 ] rozwiązaniem jest liczba 10.

30. Do budowy liczby naturalnej reprezentowanej w systemie dwójkowym możemy użyć  $A$  cyfr 1 oraz  $B$  cyfr 0, gdzie  $A, B > 0$ . Proszę napisać funkcję, która dla zadanych parametrów  $A$  i  $B$  zwraca ilość wszystkich możliwych do budowania liczb, takich że pierwsza cyfra w systemie dwójkowym (najstarszy bit) jest równa 1, a zbudowana liczba jest złożona. Na przykład dla  $A=2$ ,  $B=3$  ilość liczb wynosi 3, są to 10010(2) 10100(2) 11000(2)

31. Mamy daną liczbę całkowitą. W tablicy jednowymiarowej należy znaleźć  $n$  liczb, których suma jest równa danej liczbie. Proszę napisać funkcję `Nka`, która otrzymując jako parametry (1) tablicę `int t[N]`, (2)  $n$  (ilość liczb stanowiących sumę) oraz sumę sprawdza, ile można w niej znaleźć „enek”.

32. Dane są definicje:

```
struct punkt { float x,y,z; };  
punkt t[N];
```

Tablica `t` zawiera współrzędne  $N$  punktów leżących w przestrzeni. Punkty posiadają jednostkową masę. Proszę napisać funkcję, która sprawdza czy istnieje podzbiór punktów liczący co najmniej 3 punkty, którego środek ciężkości leży w odległości nie większej niż  $r$  od początku układu współrzędnych. Do funkcji należy przekazać tablicę `t` oraz promień  $r$ , funkcja powinna zwrócić wartość typu `bool`.

33. Dane są definicje:

```
struct punkt { float x,y; };  
punkt t[N];
```

Tablica `t` zawiera współrzędne  $N$  punktów leżących na płaszczyźnie. Punkty posiadają jednostkową masę. Proszę napisać funkcję, która sprawdza czy istnieje niepusty podzbiór  $n$  punktów, gdzie  $n < k$  oraz  $n$  jest wielokrotnością liczby 3, którego środek ciężkości leży w odległości mniejszej niż  $r$  od początku układu współrzędnych. Do funkcji należy przekazać dokładnie 3 parametry: tablicę `t`, promień  $r$ , oraz ograniczenie  $k$ , funkcja powinna zwrócić wartość typu `bool`.

34. Proszę napisać **funkcję**, która jako parametr otrzymuje liczbę naturalną i zwraca sumę iloczynów elementów wszystkich niepustych podzbiorów zbioru dzielników pierwszych tej liczby. Można założyć, że liczba dzielników pierwszych nie przekracza 20, zatem w pierwszym etapie funkcja powinna wpisać dzielniki do tablicy pomocniczej. Przykład: 60 -> [2, 3, 5] -> 2 + 3 + 5 + 2\*3 + 2\*5 + 3\*5 + 2\*3\*5 = 71

35. Dana jest tablica `int t[N]` zawierająca liczby naturalne. Proszę napisać funkcję, która odpowiada na pytanie, czy spośród (niekoniecznie wszystkich) elementów tablicy można utworzyć dwa podzbiory o jednakowej sumie elementów, tak aby suma mocy obu podzbiorów wynosiła  $k$ . Do funkcji należy przekazać wyłącznie tablicę `t` oraz liczbę naturalną  $k$ , funkcja powinna zwrócić wartość typu `bool`.

### **Zadania na ćwiczenia 9**

1. Zaimplementuj zbiór mnogościowy liczb naturalnych korzystając ze struktury listy.
  - czy element należy do zbioru
  - dodanie elementu do zbioru
  - usunięcie elementu ze zbioru
2. Zastosowania listy do implementacji tablicy rzadkiej. Proszę napisać trzy funkcje:
  - void init(node \*&tab) - inicjalizująca tablicę,
  - int value(node \*tab, int n) - zwracająca wartość elementu o indeksie n,
  - void set(node \*tab, int n, int value) - postawiająca wartość value pod indeks n.

### **Zadania na ćwiczenia 10**

3. Proszę napisać funkcję scalającą dwie posortowane listy w jedną posortowaną listę. Do funkcji należy przekazać wskazania na pierwsze elementy obu list, funkcja powinna zwrócić wskazanie do scalonej listy.
  - funkcja iteracyjna,
  - funkcja rekurencyjna.
4. Proszę napisać funkcję, która dla podanej listy odwraca kolejność jej elementów.
5. Proszę napisać funkcję, która rozdziela elementy listy wejściowej do 10 list, według ostatniej cyfry pola val. W drugim kroku powstałe listy należy połączyć w jedną listę, która jest posortowana niemalejąco według ostatniej cyfry pola val.

### **Zadania dodatkowe**

Dane są deklaracje:

```
struct node {  
    int val;  
    node* next;  
};
```

1. Proszę napisać funkcję wstawiającą na koniec listy nowy element. Do funkcji należy przekazać wskazanie na pierwszy element listy oraz wstawianą wartość.
2. Proszę napisać funkcję usuwającą ostatni element listy. Do funkcji należy przekazać wskazanie na pierwszy element listy.
3. Dana jest niepusta lista, proszę napisać funkcję usuwającą co drugi element listy. Do funkcji należy przekazać wskazanie na pierwszy element listy.
4. Dana jest niepusta lista reprezentująca liczbę naturalną. Kolejne elementy listy przechowują kolejne cyfry. Proszę napisać funkcję zwiększającą taką liczbę o 1.
5. Liczby naturalne reprezentowane jak poprzednim zadaniu. Proszę napisać funkcję dodającą dwie takie liczby. W wyniku dodawania dwóch liczb powinna powstać nowa lista.

6. Lista zawiera niepowtarzające się elementy. Proszę napisać funkcję do której przekazujemy wskaźnik na początek oraz wartość klucza. Jeżeli element o takim kluczu występuje w liście należy go usunąć z listy. Jeżeli elementu o zadanym kluczu brak w liście należy element o takim kluczu wstawić do listy.

7. Zbiór mnogościowy zawierający napisy jest reprezentowany w postaci jednokierunkowej listy. Napisy w łańcuchu są uporządkowane leksykograficznie. Proszę napisać stosowne definicje typów oraz funkcję dodającą napis do zbioru. Do funkcji należy przekazać wskaźnik do listy oraz wstawiany napis, funkcja powinna zwrócić wartość logiczną wskazującą, czy w wyniku operacji moc zbioru uległa zmianie.

9. Proszę napisać funkcję, otrzymującą jako parametr wskaźnik na pierwszy element listy o wartościach typu int, usuwającą wszystkie elementy, których wartość jest mniejsza od wartości bezpośrednio poprzedzających je elementów.

10. Proszę napisać funkcję, otrzymującą jako parametr wskaźnik na pierwszy element listy o wartościach typu int, usuwającą wszystkie elementy, których wartość dzieli bez reszty wartość bezpośrednio następujących po nich elementów.

11. Proszę napisać funkcję, która otrzymując jako parametr wskazujący na początek listy jednokierunkowej, usuwa z niej wszystkie elementy, w których wartość klucza w zapisie trójkowym ma większą ilość jedynek niż dwójek.

12. Proszę napisać funkcję, która otrzymując jako parametr wskazujący na początek listy jednokierunkowej, przenosi na początek listy te z nich, które mają parzystą ilość piątek w zapisie ósemkowym.

13. Proszę napisać funkcję, która otrzymując jako parametr wskazujący na początek listy dwukierunkowej, usuwa z niej wszystkie elementy, w których wartość klucza w zapisie binarnym ma nieparzystą ilość jedynek.

15. Proszę napisać funkcję, która pozostawia w liście wyłącznie elementy unikalne. Do funkcji należy przekazać wskazanie na pierwszy element listy.

16. Elementy w liście są uporządkowane według wartości klucza. Proszę napisać funkcję usuwającą z listy elementy o nieunikalnym kluczu. Do funkcji przekazujemy wskazanie na pierwszy element listy, funkcja powinna zwrócić liczbę usuniętych elementów.

17. Dana jest lista zawierająca ciąg obustronnie domkniętych przedziałów. Krańce przedziałów określa uporządkowana para liczb całkowitych. Proszę napisać stosowne deklaracje oraz funkcję redukującą liczbę elementów listy. Na przykład lista: [15,19] [2,5] [7,11] [8,12] [5,6] [13,17] powinien zostać zredukowany do listy: [13,19] [2,6] [7,12]

18. Kolejne elementy listy o zwiększającej się wartości pola val nazywamy podlistą rosnącą. Proszę napisać funkcję, która usuwa z listy wejściowej najdłuższą podlistę rosnącą. Warunkiem usunięcia jest istnienie w liście dokładnie jednej najdłuższej podlisty rosnącej.

19. Dana jest lista, który być może zakończona jest cyklem. Napisać funkcję, która sprawdza ten fakt.



20. Dana jest lista, który zakończona jest cyklem.  
Napisać funkcję, która zwraca liczbę elementów w cyklu.

21. Dana jest lista, który zakończona jest cyklem.  
Napisać funkcję, która zwraca liczbę elementów przed cyklem.

99. Dana jest lista, który zakończona jest cyklem. Napisać funkcję, która zwraca wskaźnik do ostatniego elementu przed cyklem.

22. Proszę napisać funkcję, która sprawdza czy jedna lista zawiera się w drugiej. Do funkcji należy przekazać wskazania na pierwsze elementy obu list, funkcja powinna zwrócić wartość logiczną.

23. Proszę napisać funkcję scalającą dwie posortowane listy w jedną posortowaną listę. Do funkcji należy przekazać wskazania na pierwsze elementy obu list, funkcja powinna zwrócić wskazanie do scalonej listy.  
- funkcja iteracyjna,  
- funkcja rekurencyjna.

24. Dwie listy zawierają niepowtarzające się (w obrębie listy) liczby naturalne. W pierwszej liście liczby są posortowane rosnąco, a w drugiej nie. Proszę napisać funkcję usuwającą z obu list liczby występujące w obu listach. Do funkcji należy przekazać wskazania na obie listy, funkcja powinna zwrócić łączną liczbę usuniętych elementów.

25. Dwie listy zawierają niepowtarzające się (w obrębie listy) liczby naturalne. W obu listach liczby są posortowane rosnąco. Proszę napisać funkcję usuwającą z każdej listy liczby nie występujące w drugiej. Do funkcji należy przekazać wskazania na obie listy, funkcja powinna zwrócić łączną liczbę usuniętych elementów.

26. Dane są dwie niepuste listy, z których każda zawiera niepowtarzające się elementy. Elementy w pierwszej liście są uporządkowane rosnąco, w drugiej elementy występują w przypadkowej kolejności. Proszę napisać funkcję, która z dwóch takich list stworzy jedną, w której uporządkowane elementy będą stanowić sumę mnogościową elementów z list wejściowych. Do funkcji należy przekazać wskazania na obie listy, funkcja powinna zwrócić wskazanie na listę wynikową. Na przykład dla list:

2 -> 3 -> 5 -> 7 -> 11

8 -> 2 -> 7 -> 4

powinna pozostać lista:

2 -> 3 -> 4 -> 5 -> 7 -> 8 -> 11

27. Proszę napisać funkcję, która rozdziela listę na dwie listy. Pierwsza powinna zawierać klucze parzyste dodatnie, drugi klucze nieparzyste ujemne, pozostałe elementy należy usunąć z pamięci. Do funkcji należy przekazać wskaźniki na listę z danymi oraz wskaźniki na listy wynikowe. Funkcja powinna zwrócić liczbę usuniętych elementów.

28. Lista reprezentuje wielomian o współczynnikach całkowitych. Elementy w liście ułożone są według rosnących potęg. Proszę napisać funkcję obliczającą różnicę dwóch dowolnych wielomianów. Wielomiany reprezentowane są przez wyżej opisane listy. Procedura powinna zwracać wskaźnik do nowo utworzonej listy reprezentującej wielomian wynikowy. Listy wejściowe powinny pozostać niezmienione.

29. Napis  $s_1$  poprzedza napis  $s_2$  jeżeli ostatnia litera  $s_1$  jest „mniejsza” od pierwszej litery  $s_2$ . Według tej zasady rozmieszczono napisy w liście cyklicznej, na przykład:

bartek—leszek—marek—ola—zosia

Proszę napisać stosowne definicje typów oraz funkcję wstawiającą do listy napis z zachowaniem zasady poprzedzania. Do funkcji należy przekazać wskaźnik do listy oraz wstawiany napis, funkcja powinna zwrócić wartość logiczną wskazującą, czy udało się wstawić napis do listy. Po wstawieniu elementu wskaźnik do listy powinien wskazywać na nowo wstawiony element.

30. Proszę napisać funkcję, która usuwa z listy cyklicznej elementy, których klucz występuje dokładnie  $k$  razy. Do funkcji należy przekazać wskazanie na jeden z elementów listy, oraz liczbę  $k$ , funkcja powinna zwrócić informację czy usunięto jakieś elementy z listy.

31. A teraz proszę rozwiązać wszystkie powyższe zadania przy założeniu że listy zawierają wartownika.

### **Zadania dodatkowe na ćwiczenia 11**

Dana jest deklaracja węzła w drzewie binarnym:

```
struct node {  
    int val;  
    node* left;  
    node* right;  
};
```

1. Proszę napisać rekurencyjne funkcje:

- Wypisującą elementy drzewa.
- Zwracającą liczbę elementów w drzewie.
- Zwracającą wysokość drzewa.
- Zwracającą liczbę liści w drzewie.
- Zwracającą liczbę węzłów na k-tym poziomie.
- Sprawdzającą czy dana wartość znajduje się w drzewie.
- Usuającą drzewo z pamięci.
- Usuającą wszystkie węzły aby pozostałe węzły tworzyły drzewo o wysokości k.

2. Proszę napisać powyższe funkcje jeszcze raz ale bez użycia rekurencji.

3. Proszę napisać funkcję wyszukiującą element w drzewie BST (w wersji rekurencyjnej i bez rekurencji)

4. Proszę napisać funkcję wstawiającą do drzewa BST nowy węzeł (w wersji rekurencyjnej i bez rekurencji)

### **WYSZUKIWANIE POŁÓWKOWE**

1. Dany jest tablica `int t[N][N]` zawierająca liczby całkowite uporządkowane niemalejąco wierszami. Proszę napisać efektywną funkcję wyznaczającą ilość liczb większych od zera.

2. Dana jest duża tablica zawierająca posortowane nazwiska. Na jaką literę jest najwięcej nazwisk?

3. Napisać funkcję zwracającą liczbę ujemnych liczb w posortowanej tablicy.

4. Dana jest posortowana tablica. Napisać funkcję wyznaczającą liczbę powtórzeń zadanej wartości w tej tablicy.