DANE DO ZADANIA

W pliku `dane.txt` znajduje się ciąg liczb całkowitych dodatnich (po jednej liczbie w każdym wierszu).

Do dyspozycji masz również mały plik testowy `przyklad.txt` (15 liczb), na którym możesz przetestować swój kod i sprawdzić zgodność z podanymi odpowiedziami.

Zawartość pliku `przyklad.txt`:

```
12
121
45
7
303
19
100
200
500
13
235
64
91
88
11
```

TREŚĆ ZADAŃ

2. **Na ocenę 2**
   Pokaż, że potrafisz wczytać dane i przejść przez nie pętlą.
   - **Zadanie 2.1**: Podaj, ile łącznie liczb znajduje się w pliku.
   - **Zadanie 2.2**: Zlicz, ile w pliku jest liczb parzystych, a ile nieparzystych.
3. **Na ocenę 3**
   Pokaż, że potrafisz wyznaczać podstawowe miary statystyczne
   - **Zadanie 3.1**: Znajdź i wypisz najmniejszą oraz największą liczbę w pliku.
   - **Zadanie 3.2**: Oblicz średnią arytmetyczną wszystkich liczb z pliku (wynik zaokrąglij do 2 miejsc po przecinku).
   - **Zadanie 3.3**: Ile jest w pliku liczb trzycyfrowych (czyli z przedziału od 100 do 999 włącznie)?

4. **Na ocenę 4**
   Pokaż, że potrafisz analizować strukturę liczby (napisy i operacje arytmetyczne).
   - **Zadanie 4.1 (Palindromy):** Podaj, ile jest w pliku liczb co najmniej dwucyfrowych, które są palindromami (czytają się tak samo od przodu i od tyłu, np. 121, 88, 303). Wypisz pierwsze trzy takie liczby napotkane w pliku.
   - Zadanie 4.2 (Suma cyfr): Ile liczb w pliku ma sumę cyfr równą dokładnie 10 (np. dla liczby 235 suma wynosi 2+3+5=10)?
5. **Na ocenę 5**
   Zastosuj algorytmy klasyczne i śledzenie sekwencji.
   - **Zadanie 5.1 (Liczby pierwsze):** Ile liczb w pliku to liczby pierwsze (liczby naturalne większe od 1, mające dokładnie dwa dzielniki: 1 i samą siebie)?
   - **Zadanie 5.2 (Rosnące serie):** Znajdź długość najdłuższego spójnego ciągu liczb rosnących występujących bezpośrednio po sobie w pliku (gdzie każda kolejna liczba jest większa od poprzedniej). Podaj:
     - długość tego ciągu,
     - pierwszą liczbę tego ciągu,
     - ostatnią liczbę tego ciągu.
6. **Na ocenę 6**
   Wykaż się znajomością optymalizacji, struktur zliczeniowych oraz teorii liczb.
   - **Zadanie 6.1 (Największy wspólny dzielnik sąsiadów):** Przejrzyj wszystkie pary liczb stojących w pliku bezpośrednio obok siebie (A_i, A_{i+1}). Znajdź parę, która ma największy wspólny dzielnik (NWD). Wypisz tę wartość NWD oraz liczby, które go tworzą.
   - **Zadanie 6.2 (Analiza częstości):** Przeanalizuj wszystkie pojedyncze cyfry we wszystkich liczbach w pliku. Która cyfra (od 0 do 9) pojawia się najczęściej i ile dokładnie razy wystąpiła?

JAK ZACZĄĆ?

1. Jak zacząć? (Struktura programu)
   1.1. Wczytaj wszystkie liczby do tablicy / listy na samym początku programu. Dzięki temu nie musisz wielokrotnie czytać pliku z dysku.
   1.2. Zanim uruchomisz kod na dużym pliku `dane.txt`, zawsze testuj na `przyklad.txt` i porównuj wyniki z tabelą kontrolną poniżej.
2. Wskazówki algorytmiczne do konkretnych zadań:
   2.1. Wskazówki algorytmiczne do konkretnych zadań:

Parzystość (Zadanie 2.2):

Liczba x jest parzysta, gdy reszta z dzielenia przez 2 wynosi 0: x % 2 === 0 (JS/C++) lub x % 2 == 0 (Python).

Min / Max (Zadanie 3.1):

Ustaw początkowe min = liczby[0] oraz max = liczby[0]. W pętli sprawdzaj: jeśli napotkasz element mniejszy od min, zaktualizuj min.

Palindrom (Zadanie 4.1):

Najprościej potraktować liczbę jako tekst!

- Python: `s = str(x); if len(s) >= 2 and s == s[::-1]: ...`
- JS: `const s = String(x); if (s.length >= 2 && s === s.split('').reverse().join('')) ...`
- C++: `string s = to_string(x); string r = s; reverse(r.begin(), r.end()); if (s.length() >= 2 && s == r) ...`

Suma cyfr (Zadanie 4.2):

Dwa podejścia:

- Zamień liczbę na napis i zsumuj wartości znaków.
- Matematycznie w pętli `while (x > 0)`: pobieraj ostatnią cyfrę przez `x % 10`, a następnie usuwaj ją przez całkowite dzielenie `x = Math.floor(x / 10)` lub `x /= 10`.

Liczba pierwsza (Zadanie 5.1):

Napisz pomocniczą funkcję `czy_pierwsza(n)`:

- n<2 – zwróć false.
- Sprawdź w pętli dzielniki d od 2 do $\lfloor\sqrt{2}\rfloor$ (`d * d <= n`).
- Jeśli n dzieli się przez którykolwiek d bez reszty (`n % d == 0`), to nie jest pierwsza (`false`). Jeśli pętla dojdzie do końca – jest pierwsza (`true`).

Ciąg rosnący (Zadanie 5.2):

Użyj dwóch liczników: `aktualna_dlugosc = 1` i `max_dlugosc = 1`
Przeglądaj tablicę od indeksu 1 do końca:

- Jeśli `liczby[i] > liczby[i - 1]` → zwiększ aktualna_dlugosc.
- W przeciwnym razie → zresetuj `aktualna_dlugosc = 1`.
- Przy każdym kroku sprawdzaj, czy `aktualna_dlugosc > max_dlugosc`. Jeśli tak, zaktualizuj rekord i zapamiętaj punkt końcowy.

NWD (Zadanie 6.1):
Wykorzystaj klasyczny algorytm Euklidesa:

```
funkcja nwd(a, b):
    dopóki b != 0:
        temp = b
        b = a % b
        a = temp
    zwróć a
```

Zliczanie cyfr (Zadanie 6.2):
Stwórz 10-elementową tablicę liczników `wystapienia = [0, 0, ..., 0]`.
Dla każdej cyfry w każdej liczbie wykonaj `wystapienia[cyfra]++`. Na końcu znajdź indeks o największej wartości.

JAK WCZYTAĆ PLIK W TWOIM JĘZYKU?

Python

```
# Wczytanie liczb do listy
with open("przyklad.txt", "r") as f:
    liczby = [int(linia.strip()) for linia in f if linia.strip()]

print(f"Wczytano {len(liczby)} liczb.")
# Tutaj pisz swój kod...
```

C++

```
#include <iostream>
#include <fstream>
#include <vector>

using namespace std;

int main() {
    ifstream plik("przyklad.txt");
    vector<int> liczby;
    int x;

    while (plik >> x) {
        liczby.push_back(x);
    }
    plik.close();

    cout << "Wczytano " << liczby.size() << " liczb." << endl;
    // Tutaj pisz swój kod...
    return 0;
}
```

ODPOWIEDZI DLA PLIKU KONTROLNEGO (`przyklad.txt`)

Zadanie	Prawidłowa odpowiedź dla `przyklad.txt`
2.1	Łącznie liczb: 15
2.2	Parzyste: 6, Nieparzyste: 9
3.1	Min: 7, Max: 500
3.2	Średnia: 120.60 (suma = 1809)
3.3	Liczby 3-cyfrowe: 6 (121, 303, 100, 200, 500, 235)
4.1	Liczba palindromów ≥ 10: 4; Pierwsze trzy: 121, 303, 88 (oraz 11)
4.2	Liczby z sumą cyfr równą 10: 4 (są to: 19, 235, 64, 91)
5.1	Liczby pierwsze: 4 (są to: 7, 19, 13, 11)
5.2	Najdłuższy ciąg rosnący: długość 4 (liczby: 19 → 100 → 200 → 500)
6.1	Największy NWD sąsiadów: 100 (dla pary 100 i 200 oraz 200 i 500)
6.2	Najczęstsza cyfra: 1 (wystąpiła 9 razy)