

**INFORMATYCZNY KONKURS TEMATYCZNY
„OD ALGORYTMU DO PROGRAMU”**

Instrukcja dla ucznia

1. Sprawdź, czy arkusz konkursowy z treścią zadań zawiera 6 stron. Ewentualny brak stron lub inne usterki zgłoś Komisji.
2. Sprawdź, czy na pulpicie utworzony został folder o nazwie zgodnej z wylosowanym przez Ciebie kodem (numerem komputera) oraz czy znajduje się w nim pliki: *zad1.xlsx*, *dec.in*, *dec1000.in*, *dec40000.in*.
3. Czytaj uważnie wszystkie teksty i zadania.
4. Rozwiązania zadań zapisuj regularnie w folderze o nazwie zgodnej z wylosowanym przez Ciebie kodem.
5. Pamiętaj o poprawnym nazywaniu plików (zgodnie z instrukcją) z rozwiązaniami zadań.
6. Rozwiązując zadania zadbaj o uniwersalność rozwiązań tak, aby działały prawidłowo **na innych zestawach danych wejściowych**.
7. Rozwiązując zadania, możesz wykorzystać wolne miejsca na arkuszu konkursowym, traktując je, jako brudnopis. Zapisy w brudnopisie nie będą sprawdzane i oceniane.
8. Po zakończeniu pracy z zadaniami, zgłoś ten fakt komisji poprzez podniesienie ręki.
9. Przedstawiciel komisji skopiuje Twoje pliki na pamięć USB.

P O W O D Z E N I A

Etap Wojewódzki

**25 marca 2019 r.
godz. 12.00**

Czas pracy:

120 minut

Liczba punktów do uzyskania:

85 punktów

ZADANIE 1. (0 – 35)

Plik źródłowy: **zad1.xlsx** (Zadanie można wykonać w programie MS Excel lub LibreOffice Calc)

Do oceny należy oddać przemianowany plik, zawierający w nazwie **Kod Ucznia**
(np. K01_zad1.xlsx lub K01_zad1.ods)

UWAGA!!! Rozwiązując zadania zadbaj o uniwersalność rozwiązań gdyż będą one sprawdzane na innym zestawie danych wejściowych umieszczonych w zakresie A1:A101 arkusza DANE. Np. NIE MOŻNA zastosować sortowania danych celem „ręcznego policzenia” ciągów znaków określonych w zadaniu i wpisania wyniku w określonej komórce. Takie rozwiązania, mimo iż będą się zgadzały dla przykładowych danych, nie będą punktowane.

W zakresie A1:A101 arkusza DANE skoroszytu **zad1.xlsx** umieszczono 100 ciągów znaków. Ciągi znaków mogą zawierać: **cyfry** od 0 do 9, **duże litery** alfabetu łacińskiego (bez polskich znaków diakrytycznych) oraz siedem **znaków specjalnych**: ! @ # \$ % ^ &. Maksymalna długość ciągu wynosi **15 znaków**. Ciągi znaków będą stanowiły dane do rozwiązania zadań z punktów od A do E.

Dane do zadania wyróżniono szarym kolorem tła – nie należy ich zmieniać. W pozostałych arkuszach wyróżniono żółtym kolorem miejsca, w których należy umieścić rozwiązania. **Wszelkie dodatkowe obliczenia pomocnicze należy umieścić bezpośrednio w arkuszach, których dotyczą - nie należy ich ukrywać.**

Na podstawie tych informacji wykonaj poniższe polecenia za pomocą odpowiednich formuł, funkcji i funkcjonalności jakie oferuje arkusz kalkulacyjny.

A. (0 – 5)

Oblicz ile spośród ciągów znaków przedstawia **poprawne liczby zapisane w heksadecymalnym (szesnastkowym) systemie liczbowym**. Odpowiedź umieść w wyróżnionej komórce C5 arkusza **PunktA**.

B. (0 – 5)

Oblicz ile spośród ciągów znaków **składa się wyłącznie z liter**. Odpowiedź umieść w wyróżnionej komórce C5 arkusza **PunktB**.

C. (0 – 7)

Oblicz ile spośród ciągów znaków **jest palindromami** (*palindromem jest ciąg znaków, który czytany od lewej i prawej strony jest taki sam, np. OKO, ALU#8778#ULA*). Odpowiedź umieść w wyróżnionej komórce C5 arkusza **PunktC**.

D. (0 – 6)

Oblicz **łącną ilość liter, cyfr i znaków zdefiniowanych w treści zadania jako specjalne** we wszystkich ciągach znaków. Odpowiedzi umieść w wyróżnionych komórkach C5:C7 arkusza **PunktD**.

E. (0 – 12)

Na podstawie danych zawartych w arkuszu DANE sporządź **wykres kołowy** przedstawiający:

- ilości liczb heksadecymalnych parzystych
- ilości liczb heksadecymalnych nieparzystych
- ilości ciągów znaków, które nie są poprawnymi liczbami heksadecymalnymi.

Wykres powinien zawierać:

- tytuł wykresu – *Struktura danych*
- legendę zawierającą trzy pozycje (liczby parzyste, liczby nieparzyste, znaki specjalne)
- etykiety danych w formacie procentowym

Dane do wykresu przygotuj i umieść w arkuszu **PunktE** natomiast wykres umieść w odrębnym arkuszu o nazwie **Wykres**.

ZADANIE 2. (0 – 50)

Plik z przykładowymi danymi wejściowymi: **dec.in**

Plik wyjściowy: **dec.out**

Dodatkowe pliki do testowania rozwiązań z większą liczbą danych: **dec1000.in, dec40000.in**

Do oceny należy oddać plik źródłowy z rozwiązaniem zadania, zawierający w nazwie **Kod Ucznia**

(np. K01_zad2.cpp lub K01_zad2.c)

NA POTRZEBY ZADANIA WPROWADZONO NASTĘPUJĄCE DEFINICJE

- Liczba parzysta** - to liczba podzielna przez 2 bez reszty (np. liczba **14**)
- Liczba C-parzysta** - to liczba, która jest **parzysta** oraz składa się tylko i wyłącznie z **cyfr parzystych** czyli cyfr: **0, 2, 4, 6, 8**. (np. liczba **22**)
- Liczba B-parzysta** - to liczba, która jest **C-parzysta** oraz suma cyfr tej liczby zapisanej w binarnym systemie liczbowym jest **parzysta**. (np. liczba **222** czyli 11011110_2)
- Liczba Super-B-parzysta** - to liczba, która jest **B-parzysta** oraz ilości cyfr "1" i cyfr "0" tej liczby zapisanej w binarnym systemie liczbowym są sobie równe. (np. liczba **202** czyli 11001010_2)

Opis organizacji danych w pliku wejściowym

W pierwszym wierszu pliku **dec.in** jest zapisana jedna liczba naturalna **n** mówiąca o ilości liczb zapisanych w kolejnych wierszach pliku. Liczba **n** mieści się w przedziale domkniętym od **1** do **40000**. W każdym z kolejnych **n** wierszy zapisana jest jedna liczba całkowita mieszcząca się w przedziale domkniętym od **1** do **100000**. Liczby te, to **dane do rozwiązania zadania**. Zapisane są one w pliku wejściowym w losowej kolejności i mogą się powtarzać.

Przykład organizacji danych pliku *dec.in*:

```
5
999
222
22
202
999
```

Korzystając z powyższych informacji napisz program, który wykona czynności wymienione w punktach od A do E (przy każdym z nich podano maksymalną punktację możliwą do uzyskania):

A. (0 - 10)

Wczytaj liczby całkowite z pliku wejściowego do odpowiednich struktur danych oraz umieść w kodzie źródłowym w postaci komentarzy nazwy zmiennych wraz z typami danych jakie zostały użyte do zapamiętania powyższych informacji. Wyświetl na ekranie i zapisz w pliku wyjściowym **ilość liczb większych bądź równych** od średniej arytmetycznej wszystkich liczb stanowiących dane do rozwiązania zadania.

B. (0 - 9)

Oblicz **która z liczb** powtarza się w pliku wejściowym największą ilość razy. Najczęściej powtarzającą się liczbę wyświetl na ekranie i zapisz do pliku wyjściowego. (Uwaga. Jest tylko jedna taka liczba)

C. (0 - 9)

Oblicz **ilość liczb C-parzystych** w pliku wejściowym. Wynik wyświetl na ekranie i zapisz do pliku wyjściowego.

D. (0 - 10)

Oblicz **ilość liczb B-parzystych** w pliku wejściowym. Wynik wyświetl na ekranie i zapisz do pliku wyjściowego.

E. (0 - 10)

Oblicz **ilość liczb Super-B-parzystych** w pliku wejściowym. Wynik wyświetl na ekranie i zapisz do pliku wyjściowego.

Uwaga: Za poprawny zapis zestawu wyników w pliku wyjściowym *dec.out* można uzyskać dodatkowo 2 pkt.

UWAGI DO ROZWIĄZAŃ

- Dane zawarte w przykładowych plikach wejściowych *dec.in*, *dec1000.in*, *dec40000.in* są poprawne (nie wymagają sprawdzania poprawności).
- W rozwiązaniach należy zadbać o odpowiednie dobranie struktur i typów danych do przechowywanych zawartości, poprawną inicjalizację zmiennych, czytelność i przejrzystość kodu i zastosowanie algorytmu o jak najmniejszej złożoności czasowej.
- Rozwiązania będą testowane i oceniane na innych danych niż załączone do zadania (zgodnych ze specyfikacją zadania).
- Do oceny należy oddać tylko plik źródłowy (z rozszerzeniem **cpp** lub **c**), realizujący rozwiązanie zadań z punktów od A do E, nie należy załączać pliku wykonywalnego **exe**.
- Program powinien składać się tylko z jednego pliku i korzystać jedynie ze standardowych bibliotek (*iostream*, *fstream*, *cstdlib*, *stdio.h*, *cmath*).
- Dane do programu powinny być pobierane z pliku wejściowego. Można to wykonać poprzez **wczytanie danych z pliku wejściowego** lub poprzez **przekierowanie standardowego wejścia**.
- Rozwiązania nie mogą tworzyć plików tymczasowych, procesów czy wątków, korzystać z funkcji sieciowych, oczekiwać na interakcję użytkownika, uruchamiać innych programów, korzystać z zewnętrznych bibliotek (oprócz wymienionych powyżej).
- Przed rozwiązaniami punktów zadania należy wyświetlić na ekranie i zapisać w pliku wyjściowym odpowiednią literę określającą poszczególne części zadania.

Przykład informacji, jaka powinna zostać zapisana w pliku wyjściowym oraz pojawić się na ekranie po uruchomieniu programu:

Dla pliku wejściowego *dec.in* o zawartości jak w opisie zadania, program powinien wyświetlić na ekranie i zapisać w pliku wyjściowym *dec.out* następujące informacje:

A

2

B

999

C

3

D

2

E

1