Zadanie 12: Pętle

Treść zadania

Napisz sześć skryptów JavaScript. Wyniki wypisz do konsoli (`console.log`).

1. Wypisz 100 pierwszych liczb naturalnych podzielnych przez 7.

2. Wypisz kolejne liczby naturalne podzielne przez 2 w kolejności malejącej z przedziału od 80 do 120.

3. Wyznacz, ile kolejnych liczb naturalnych, rozpoczynając od 10, trzeba dodać do siebie, aby ich suma wynosiła co najmniej 1000. Wypisz również, ile wynosiła wartość ostatniej dodawanej liczby.

4. Pobierz od użytkownika liczbę i wypisz wartość jej silni. Wyjaśnij, czym jest silnia.

5. Wypisz tekst składający się z 20 gwiazdek:

```
********************
```

6. Wypisz tekst złożony z 5 wierszy po 20 gwiazdek każdy. Do przejścia do nowego wiersza użyj znaku `\n`.

```
********************
********************
********************
********************
********************
```

Przyjęte założenia. W zadaniu 1 liczby naturalne zaczynają się od 1, więc wynik to 7, 14, …, 700. Jeśli do liczb naturalnych zaliczasz także 0, ciąg zaczyna się od 0 i kończy na 693. W zadaniu 2 końce przedziału (80 i 120) wchodzą do wyniku.

Wskazówki

- `for` stosujemy, gdy znamy liczbę powtórzeń, a `while`, gdy powtarzamy do spełnienia warunku,
- w pętli `for` krok może być inny niż 1 (`i += 2`) i może być ujemny (`i--`),
- zapis `suma += liczba` to skrót od `suma = suma + liczba`, a `silnia *= i` od `silnia = silnia * i`,
- tekst można budować w pętli, dopisując znaki operatorem `+=`,
- znak `\n` w tekście oznacza przejście do nowego wiersza.

Czym jest silnia?

Silnia liczby naturalnej n (zapis `n!`) to iloczyn wszystkich liczb naturalnych od 1 do n:

```
n! = 1 * 2 * 3 * ... * n
```

- Przykłady: `3! = 1 * 2 * 3 = 6`, `5! = 120`, `10! = 3628800`.
- Z definicji `0! = 1` (i `1! = 1`).
- Silnia nie jest określona dla liczb ujemnych ani ułamkowych (w tym zadaniu).
- Zależność: `n! = n * (n - 1)!`, np. `5! = 5 * 4!`.
- Zastosowanie: silnia mówi, na ile sposobów można ustawić n różnych elementów w kolejności. Trzy osoby ustawimy w kolejce na `3! = 6` sposobów.

- Silnia rośnie bardzo szybko. W JavaScript liczby większe niż około $1{,}8 \cdot 10^{308}$ są zapisywane jako `Infinity`, co dotyczy już `171!`.

Rozwiązanie

Pytania teoretyczne

Pytanie 1. Czym różnią się pętle `for`, `while` i `do...while`? Którą wybierzesz w zadaniu 2, a którą w zadaniu 3 i dlaczego?

Pytanie 2. Ile razy wykona się pętla `for (let i = 120; i >= 80; i -= 2)` i jaka będzie ostatnia wartość `i` wewnątrz pętli?

Pytanie 3. Dlaczego pętla `while (suma < 1000)`, w której nie zmieniamy `suma` ani `liczba`, nigdy się nie zakończy? Jak się nazywa taka sytuacja?

Pytanie 4. Dlaczego zmienną `silnia` ustawiamy na 1, a nie na 0? Ile wynosi `0!`?

Pytanie 5. Co wypisze poniższy kod?

```
let s = "";
for (let i = 0; i < 3; i++) {
    s += i;
}
console.log(s);
```

A. 3

B. 012

C. 6

D. błąd

Pytanie 6. Co wypisze `console.log("a\nb");`? Dlaczego w zadaniu 6 nie dopisujemy `\n` po ostatnim wierszu?

Odpowiedzi do pytań teoretycznych