

8

Pętle

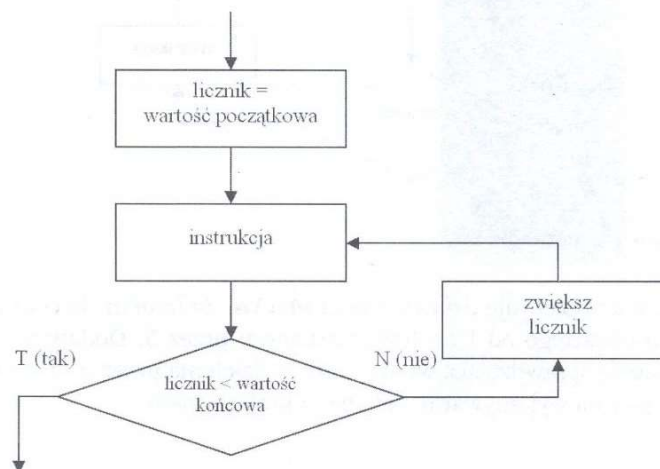
ZAGADNIENIA

- Jakie zadanie mają pętle?
- Jakie pętle są dostępne w języku JavaScript?
- Jak stosować pętle w skrypcie?
- Jak stosować instrukcję warunkową wewnątrz pętli?

Język JavaScript oferuje trzy rodzaje **pętli** (instrukcji iteracyjnych), których zadaniem jest powtarzanie instrukcji określoną liczbę razy. Pętla **for** jest najczęściej stosowanym rodzajem pętli. Składnia pętli jest następująca:

```
for (wyrażenie początkowe; warunek; wyrażenie modyfikujące){  
  instrukcje;  
}
```

Wyrażenie początkowe odpowiada za zainicjowanie zmiennej używanej jako licznik przebiegu pętli. Spełnienie warunku umożliwia wykonanie kolejnego przejścia pętli. Wyrażenie modyfikujące dostosowuje zmienną będącą licznikiem pętli. Pętlę **for** można przedstawić również w postaci schematu blokowego (rys. 8.1).

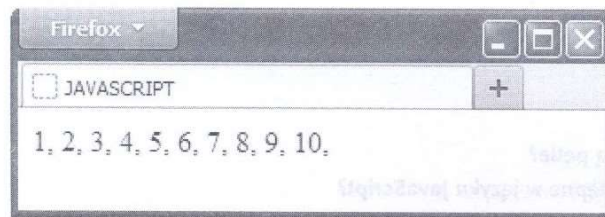


Rys. 8.1. Schemat blokowy pętli **for**

Przykład z list. 8.1 prezentuje wykorzystanie pętli **for**. Za jej pomocą dziesięciokrotnie wykonana została instrukcja **document.write**. Końcowym efektem jest wypisanie w oknie przeglądarki (rys. 8.2) cyfr od 1 do 10.

Listing 8.1

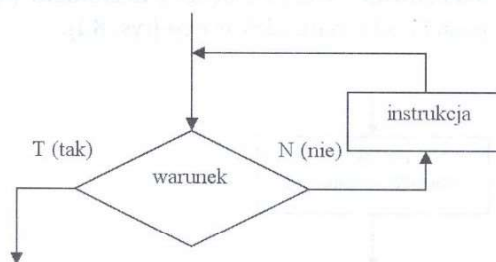
```
<script type="text/javascript">
for (x=1;x<=10;x++)
document.write(x+", ")
</script>
```

Rys. 8.2. Wynik działania skryptu z list. 8.1 – pętla **for**

Pętla **while** przed wykonaniem instrukcji sprawdza warunek logiczny. Jeżeli warunek przyjmuje wartość **true**, pętla będzie wykonywana do czasu osiągnięcia przez warunek wartości **false**. Może zdarzyć się sytuacja, że pętla nie wykona się ani razu, gdy za pierwszym razem warunek przyjmie wartość **false**. Składnia pętli jest następująca:

```
while (warunek) {
instrukcje;
}
```

Pętlę **while** w postaci schematu blokowego prezentuje rys. 8.3.

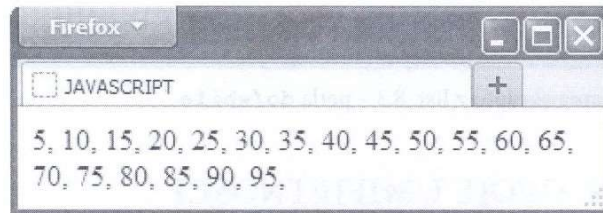
Rys. 8.3. Schemat blokowy pętli **while**

Przykład z list. 8.2 prezentuje działanie pętli **while**. Zadaniem skryptu jest wypisanie liczb z przedziału otwartego od 1 do 100 podzielnych przez 5. Dodatkowo zastosowano instrukcję warunkową sprawdzającą wynik reszty z dzielenia przez 5 oraz operator inkrementacji pozwalający na wykonywanie kolejnego kroku w pętli.

Listing 8.2

```
<script type="text/javascript">
var x=1;
while(x<100)
{
if (x%5==0)
```

```
document.write(x+", ");
x++;
}
</script>
```

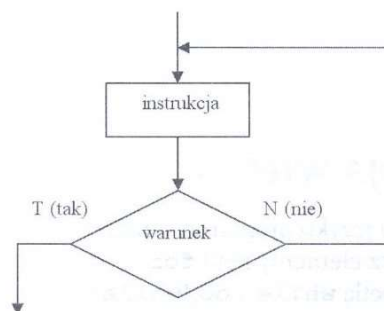


Rys. 8.4. Wynik działania skryptu z list. 8.2 – pętla **while**

Pętla **do/while** pozwala na wykonanie instrukcji przynajmniej raz, zanim zostanie sprawdzony warunek logiczny. Składnia pętli jest następująca:

```
do {
instrukcje;
}
while (warunek)
```

Pętlę **do/while** w postaci schematu blokowego prezentuje rys. 8.5.

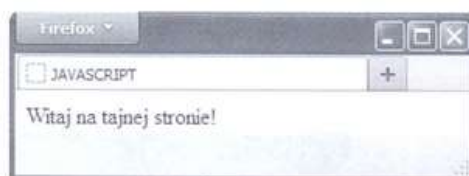


Rys. 8.5. Schemat blokowy pętli **do/while**

Przykład list. 8.3 prezentuje działanie pętli **do/while**. Skrypt powtarza wyświetlanie okna tekstowego. Warunkiem przejścia do strony jest wpisanie w oknie tekstowym hasła **JavaScript**.

Listing 8.3

```
<script type="text/javascript">
do{
var haslo=prompt("Podaj hasło", "")
}
while (haslo!="JavaScript")
document.write("Witaj na tajnej stronie!")
</script>
```



Rys. 8.6. Wynik działania skryptu z list. 8.3 – pętla `do/while`

SPRAWDŹ SVOJE UMIEJĘTNOŚCI

1. Stosując pętle i jedną instrukcję `document.write("#")`; , wyświetl w przeglądarce następujące struktury:

a) linię składającą się z ośmiu #:

```
#####
```

b) trójkąt o dolnej podstawie pięciu #:

```
#
```

```
##
```

```
###
```

```
####
```

```
#####
```

c) prostokąt zbudowany ze znaków # o wymiarach 3x7:

```
#####
```

```
#####
```

```
#####
```

SPRAWDŹ SWOJĄ WIEDZĘ

1. Za co odpowiadają pętle w języku programowania?
2. Przedstaw strukturę i opisz elementy pętli `for`.
3. Jaka jest różnica między pętlą `while` a `do/while`?

ZADANIA

Utwórz skrypt, którego zadaniem jest

- a) wyznaczenie n elementów ciągu Fibonacciego ($n > 0$)
- b) sprawdzenie czy podana liczba całkowita n jest pierwsza
- c) wyznaczenie rozkładu liczby $n > 1$ na czynniki pierwsze
- d) wyznaczenie liczb podzielnych przez 3 z przedziału od a do b
- e) wypisanie liczb podzielnych przez 3 i 5 jednocześnie z przedziału od 1 do n ($1 < n < 1000$)
- f) wypisanie wszystkich liczb postaci $3^n + 2$ z przedziału (1; 1000>
- g) wypisanie liczb pierwszych z przedziału (1; 1000> nie zawierających cyfry 3.