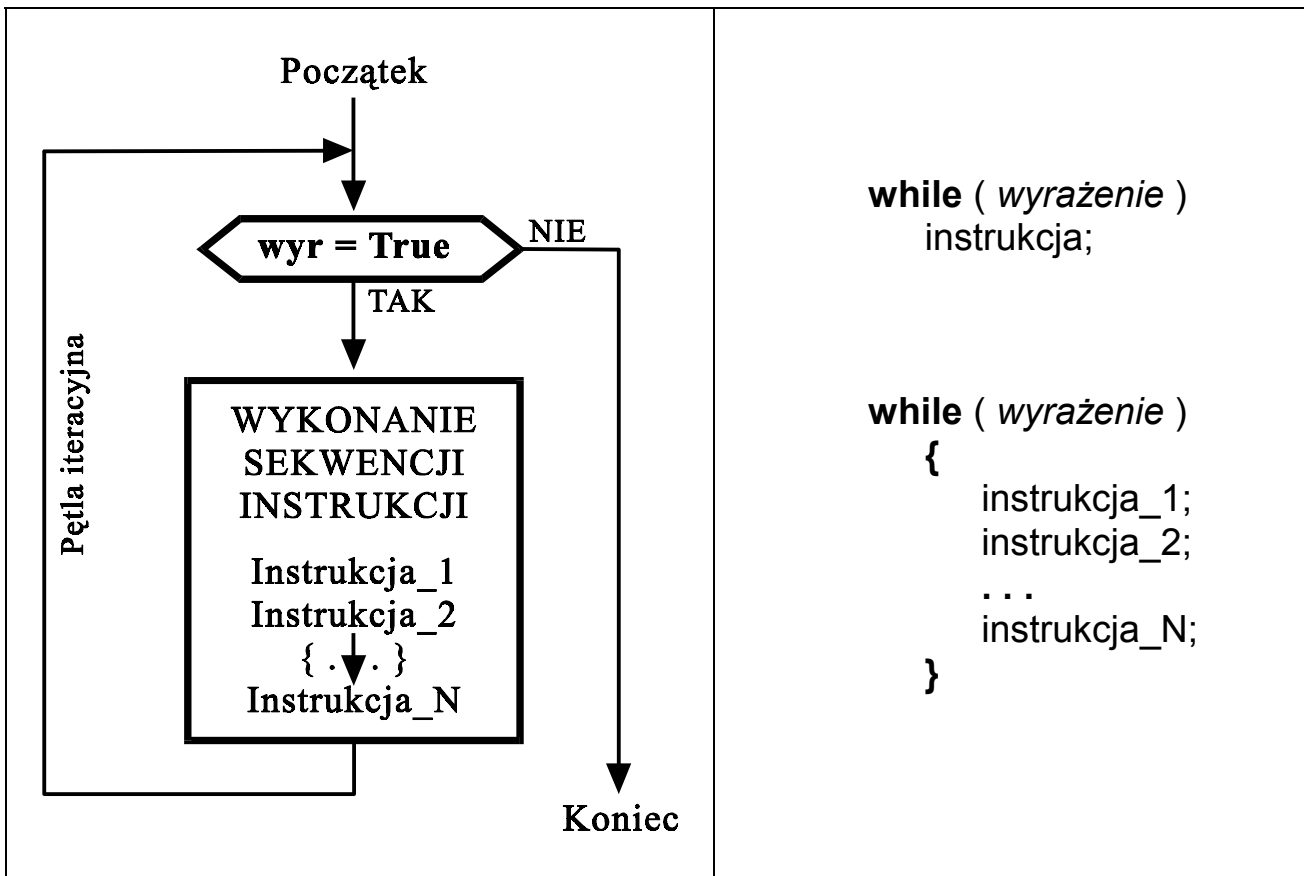


INSTRUKCJE REPETYCYJNE – PĘTLE

- Pętla **while()**:

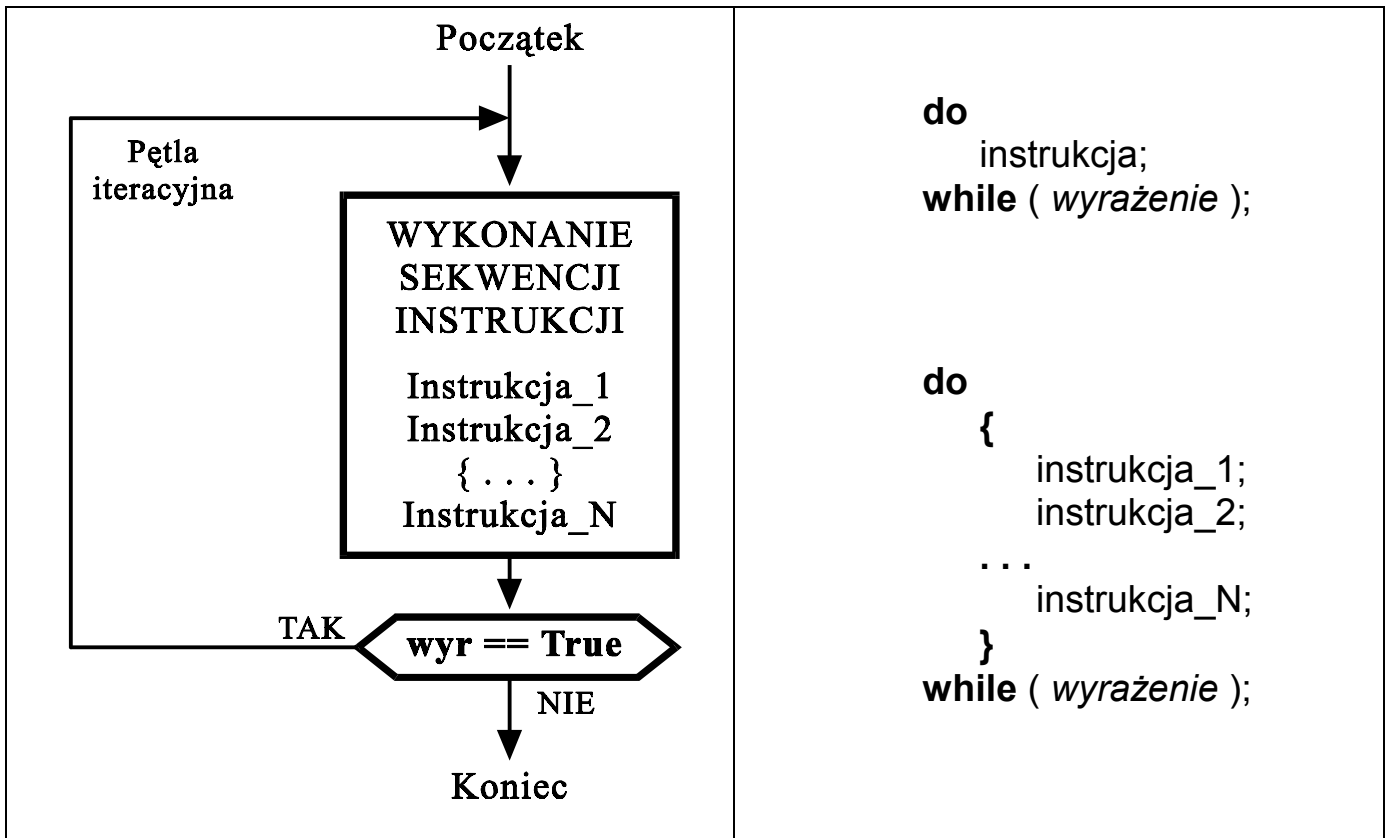


Pętla wykonywana jest tak długo jak wartość **wyrażenie** jest różna od zera

Przykłady:

<pre>int i ; // pętla wyświetlająca liczby 1,2,3 ... i = 1; while(i <=10) { printf ("%2d\n" , i); i = i + 1; }</pre>	<pre>int i = 1; // 1, 2, 3, ... w innym zapisie while(i < 11) printf ("%2d\n" , i++);</pre>
<pre>int i ; // pętla wyświetlająca liczby 10,9,8 i = 10; while(i != 0) { printf ("%2d\n" , i); i = i - 1; }</pre>	<pre>int i = 10; //10, 9, 8, ... w innym zapisie while(i) printf ("%2d\n" , i --);</pre>

- Pętla **do while()**:



Pętla wykonywana jest tak długo jak wartość **wyrażenie** jest różna od zera

Przykłady:

<pre> int i ; // pętla wyświetlająca liczby 1,2,3 ... i = 1; do { printf ("%2d\n" , i); i = i + 1; } while(i<=10); </pre>	<pre> int i = 1; // 1, 2, 3, ... w innym zapisie do printf ("%2d\n" , i); while(++i <11); </pre>
<pre> int i ; // pętla wyświetlająca liczby 10,9,8 i = 10; do { printf ("%2d\n" , i); i = i - 1; } while(i != 0); </pre>	<pre> int i = 10; // 10, 9, 8, ... w innym zapisie do printf ("%2d\n" , i); while(--i); </pre>

Inne przykłady:

```
#include <stdio.h>
#include <conio.h>

void przyklad_1( ) //odczytywanie klawiszy do momentu naciśnięcia 'k'
{
clrscr(); printf( "Przyklad_1: ilustracja petli <do while>, koniec po naciśnięciu 'k' " );
char znak;
do
{ // za pomocą petli "do while"
printf( " \npodaj znak: " );
znak = getche( );
}
while( znak != 'k' );

printf( " \n\nTeraz przyklad wykorzystania petli <while> " );
znak = 0;
while( znak != 'k' )
{
printf( " \npodaj znak: " );
znak = getche( );
}
printf( " \nWykryto naciśnięcie klawisza 'k' " ); getch( );
} //przyklad_1
```

```
#include <stdlib.h> //dodanie biblioteki zawierającej funkcje "random"
void przyklad_2( ) //odgadywanie wartości wylosowanej liczby z przedziału 1-6
{
int liczba_losowa, liczba_wczytana;
liczba_losowa = 1 + random(6); // losowanie liczby
clrscr(); printf( " Przyklad_2: losowanie i odgadywanie liczby z przedziału <1,6> " );
do // rozwiązanie z wykorzystaniem petli "do while"
{
printf("\nOdgadnij wylosowaną liczbę: ");
scanf("%d",&liczba_wczytana);
}
while( liczba_wczytana != liczba_losowa );
printf( " \nOdgadles ! Wylosowana liczba to: %d " , liczba_losowa );

printf( " \n\nJeszcze raz to samo z wykorzystaniem petli <while> " );
liczba_wczytana = 0;
while( liczba_wczytana != liczba_losowa )
{
printf( " \nOdgadnij wylosowaną liczbę: " );
scanf( "%d" , &liczba_wczytana);
}
printf( " \nOdgadles ! Wylosowana liczba to: %d " , liczba_losowa); getch();
} // przyklad_2
```

- Pętla for():

```
for( wyrażenie_inicjujace ; wyrażenie_testujace ; wyrażenie_modyfikujace )
    wykonywana_instrukcja ;
```

jest równoważna konstrukcji:

```
wyrażenie_inicjujace ;
while( wyrażenie_testujace )
{
    wykonywana_instrukcja ;
    wyrażenie_modyfikujace ;
}
```

<pre>int i ; i = 10; while(i != 0) { printf ("%2d\n" , i); i = i - 1; }</pre>	<pre>int i ; for(i = 10; i != 0 ; i = i - 1) printf("%2d\n" , i); lub int i ; for(i = 10; i ; printf("%2d\n" , i --)) ;</pre>
---	--

//przykładowy program wypisujący tabele kodów ASCII

```
#include <stdio.h>
void main(void)
{
    int n;
    printf( "\n" );
    for( n=32; n<256; n++ )
        printf( "%3d = %c\n" , n , n );
}
```

//prymitywny kalkulator biurowy

```
#include <stdio.h>
void main(void)
{
    double suma=0, liczba;
    while( scanf( "%lf" , &liczba ) > 0 )
        printf( "\t%.2f\n" , suma+=liczba );
}
```

```

#include <stdio.h> // program zliczający naciskane klawisze
#include <conio.h>
void main(void)
{
    int licznik = 0, klawisz;
    printf( "Program zliczający naciskane klawisze. Koniec = ESC" );
    do
    {
        klawisz = getch();
        licznik++;
    }
    while( klawisz != 27 ) // 27 = kod klawisza Escape
    printf( "\n Ilość naciśniętych klawiszy = %d" , licznik );
}

```

```

... // ten sam program z użyciem pętli for
int licznik;
for( licznik = 0 ; getch() != 27 ; licznik++ )
...

```

```

#include <stdio.h> // program klasyfikujący naciskane klawisze
#include <conio.h>
#define ESC 27 // definicja kodu klawisza «Escape»
void main(void)
{
    int klawisz=0;
    clrscr();
    while( klawisz != ESC )
    {
        printf( "\n\nNaciśnij jakiś klawisz (ESC->Koniec): " );
        klawisz = getch();
        if( 'a'<=klawisz && klawisz<='z' )
            printf( "-> To jest mala litera." );
        else if( 'A'<=klawisz && klawisz<='Z' )
            printf( "-> To jest duza litera." );
        else if( '0'<=klawisz && klawisz<='9' )
            printf( "-> To jest cyfra." );
        else if( klawisz == 13 )
            printf( "-> To jest klawisz ENTER." );
        else if( klawisz == ' ' )
            printf( "-> To jest klawisz spacji" );
        else
            printf( "-> To jest inny klawisz." );
    }
}

```

```

#include <stdio.h>                                //program rozpoznajacy klawisze funkcyjne
#include <conio.h>
#include "def_klaw.h"                            //dołaczenie pliku zawierajacego definicje klawiszy
void main( void )
{
    int klawisz;
    clrscr();
    do
    {
        printf( "\n\n Nacisnij jakis klawisz: " );
        klawisz = getch( );
        switch( klawisz )
        {
            case ENTER : printf( "To jest ENTER" ); break;
            case ESC    : printf( "To jest ESCAPE" ); break;
            case ZERO   : // pierwszy odczytany znak mial kod rowny 0
                klawisz = getch( );
                switch( klawisz )
                {
                    case DELETE      : printf( "Delete" ); break;
                    case UP_ARROW    : printf( "Up arrow" ); break;
                    case DOWN_ARROW  : printf( "Down arrow" ); break;
                }
                break;
            case BACKSPACE : printf( "To jest BACKSPACE" ); break;
            default       : printf( "Nieznany pojedynczy klawisz" ); break;
        }
    }
    while( klawisz != ESC );
}

```

```

// Zbior «def_klaw.h» zawierajacy definicje kodow klawiszy
#ifndef DEF_KLAW
#define DEF_KLAW
#define ZERO 0
// klawisze "zwykle" - kodowane za pomoca jednego znaku
#define ESC 27
#define ENTER 13
#define BACKSPACE 8
//klawisze "funkcyjne" - kodowane za pomoca dwoch znakow
#define DELETE 83 // 0, 83
#define UP_ARROW 72 // 0, 72
#define DOWN_ARROW 80 // 0, 80
#define LEFT_ARROW 75 // 0, 75
#define RIGHT_ARROW 77 // 0, 77
#define HOME 71 // 0, 71
#define END 79 // 0, 79
#endif

```