



Wydział Elektryczny
Katedra Elektrotechniki Teoretycznej i Metrologii

Materiały do wykładu z przedmiotu:
Informatyka
Kod: EDS1B1007

WYKŁAD NR 1

Opracował: **dr inż. Jarosław Forenc**
Białystok 2019

Materiały zostały opracowane w ramach projektu „PB2020 - Zintegrowany Program Rozwoju Politechniki Białostockiej” realizowanego w ramach Działania 3.5 Programu Operacyjnego Wiedza, Edukacja, Rozwój 2014-2020 współfinansowanego ze środków Europejskiego Funduszu Społecznego.

Dane podstawowe

- dr inż. Jarosław Forenc
- Politechnika Białostocka, Wydział Elektryczny,
Katedra Elektrotechniki Teoretycznej i Metrologii
ul. Wiejska 45D, 15-351 Białystok
WE-204
- e-mail: j.forenc@pb.edu.pl
- tel. (0-85) 746-93-97
- <http://jforenc.prv.pl>
 - Dydaktyka - slajdy prezentowane na wykładzie
- Konsultacje
 - wtorek, godz. 10:00-11:30, WE-204
 - piątek, godz. 10:00-11:30, WE-204
 - sobota, godz. 14:30-16:00, WE-204

Program wykładu (1/2)

1. **Programowanie w języku C.** Deklaracje i typy zmiennych, operatory i wyrażenia arytmetyczne, operacje wejścia-wyjścia.
2. **Programowanie w języku C.** Operatory relacyjne i logiczne, wyrażenia logiczne, instrukcja warunkowa if, instrukcja wyboru wielowariantowego switch, operator warunkowy, pętle (for, while, do .. while).
3. **Programowanie w języku C.** Tablice jedno- i dwuwymiarowe, łańcuchy znaków, struktury, wskaźniki, dynamiczny przydział pamięci.
4. **Programowanie w języku C.** Funkcje użytkownika, przekazywanie argumentów do funkcji, rekurencyjne wywołanie funkcji, pliki tekstowe i binarne.

Program wykładu (2/2)

5. **Algorytmy komputerowe.** Definicja algorytmu. Klasyfikacje, sposoby przedstawiania i złożoność obliczeniowa algorytmów.
6. **Budowa i zasada działania komputera.** Procesor, pamięć wewnętrzna i zewnętrzna. Komunikacja z urządzeniami zewnętrznymi, interfejsy komputerowe.
7. **System operacyjny.** Zarządzanie procesami, pamięcią i dyskowymi operacjami wejścia-wyjścia (systemy plików).
8. Zaliczenie wykładu.

Literatura

1. S. Prata: Język C. Szkoła programowania. Wydanie VI. Helion, 2016.
2. B.W. Kernighan, D.M. Ritchie: Język ANSI C. Programowanie. Wydanie II. Helion, Gliwice, 2010.
3. R. Reese: Wskaźniki w języku C. Przewodnik. Helion, Gliwice, 2014.
4. R. Kawa, J. Lembas: Wykłady z informatyki. Wstęp do informatyki. PWN, Warszawa 2017.
5. P. Wróblewski: Algorytmy, struktury danych i techniki programowania. Wydanie VI. Helion, Gliwice, 2019.
6. W. Stallings: Systemy operacyjne. Architektura, funkcjonowanie i projektowanie. Wydanie IX. Helion, Gliwice, 2018.
7. G. Coldwin: Zrozumieć programowanie. PWN, Warszawa, 2015.
8. A.S. Tanenbaum, H. Bos: Systemy operacyjne. Wydanie IV. Helion, Gliwice, 2015.

Efekty uczenia się i system ich oceniania

Podstawę do zaliczenia przedmiotu (uzyskanie punktów ECTS) stanowi stwierdzenie, że **każdy** z założonych **efektów kształcenia (uczenia się)** został osiągnięty w co najmniej minimalnym akceptowalnym stopniu.

EU1	identyfikuje i opisuje zasadę działania podstawowych elementów systemu komputerowego oraz charakteryzuje podstawowe zadania systemu operacyjnego
EU2	formułuje algorytmy komputerowe rozwiązujące typowe zadania inżynierskie występujące w elektrotechnice

Zaliczenie wykładu - efekty uczenia się (EU1)

- Student, który zaliczył przedmiot:

identyfikuje i opisuje zasadę działania podstawowych elementów systemu komputerowego oraz charakteryzuje podstawowe zadania systemu operacyjnego

- Student, który zalicza na ocenę **dostateczny (3)**:
 - wymienia podstawowe elementy systemu komputerowego i podaje ich przeznaczenie
 - wyjaśnia podstawowe pojęcia związane z architekturą i zasadą działania systemów komputerowych
 - podaje definicję i wymienia podstawowe zadania systemu operacyjnego
 - opisuje wybraną metodę przydziału pamięci dyskowej



Zaliczenie wykładu - efekty uczenia się (EU1)

- Student, który zalicza na ocenę **dobry (4)** (oprócz wymagań na ocenę 3):
 - opisuje strukturę i zasadę działania wybranych elementów systemu komputerowego
 - wymienia różnice pomiędzy architekturą von Neumana i architekturą harwardzką systemów komputerowych
 - podaje strukturę dysku logicznego w wybranym systemie plików (FAT, NTFS, ext)
 - wyjaśnia pojęcia stronicowania i segmentacji pamięci oraz opisuje zasadę działania pamięci wirtualnej

Zaliczenie wykładu - efekty uczenia się (EU1)

- Student, który zalicza na ocenę **bardzo dobry (5)** (oprócz wymagań na ocenę 4):
 - przedstawia cel stosowania oraz zasadę działania pamięci podręcznej
 - opisuje sposób przechowywania informacji o położeniu pliku na dysku w wybranym systemie plików (FAT, NTFS, ext)

Zaliczenie wykładu - efekty uczenia się (EU2)

- Student, który zaliczył przedmiot:

formułuje algorytmy komputerowe rozwiązujące typowe zadania inżynierskie występujące w elektrotechnice

- Student, który zalicza na ocenę **dostateczny (3)**:
 - przedstawia rozwiązanie prostego problemu w postaci schematu blokowego opisującego algorytm komputerowy
 - podaje definicję algorytmu komputerowego i wymienia metody opisu algorytmów
 - przedstawia sposób sortowania wektora liczb stosując wybraną, prostą metodę sortowania

Zaliczenie wykładu - efekty uczenia się (EU2)

- Student, który zalicza na ocenę **dobry (4)** (oprócz wymagań na ocenę 3):
 - przedstawia rozwiązanie złożonego problemu w postaci schematu blokowego opisującego algorytm komputerowy
 - wyjaśnia pojęcie złożoności obliczeniowej algorytmu, podaje złożoności obliczeniowe przykładowych algorytmów
- Student, który zalicza na ocenę **bardzo dobry (5)** (oprócz wymagań na ocenę 4):
 - wyjaśnia pojęcie rekurencji i podaje przykłady algorytmów rekurencyjnych

Zaliczenie wykładu

- Sprawdzian pisemny na koniec semestru:
 - sprawdzian: ostatni wykład w semestrze
 - poprawa: termin do ustalenia (sesja egzaminacyjna)
- Na zaliczeniu oceniane będą dwa efekty uczenia się (EU1, EU2)
- Za każdy efekt uczenia się można otrzymać od 0 do 100 pkt.
- Na podstawie otrzymanych punktów wystawiana jest ocena:

Punkty	Ocena	Punkty	Ocena
91 - 100	5,0	61 - 70	3,5
81 - 90	4,5	51 - 60	3,0
71 - 80	4,0	0 - 50	2,0

Zaliczenie wykładu

- Ocena końcowa wyznaczana jest na podstawie sumy otrzymanych punktów:

Punkty	Ocena	Punkty	Ocena
182 - 200	5,0	122 - 141	3,5
162 - 181	4,5	102 - 121	3,0
142 - 161	4,0	0 - 101	2,0

Plan wykładu nr 1

- Historia języka C
- Struktura programu, zapis kodu, sekwencje sterujące
- Komentarze

Język C - krótka historia (1/2)

- **1969** - język BCPL - Martin Richards, University Mathematical Laboratories, Cambridge
- **1970** - język B - Ken Thompson, adaptacja języka BCPL dla pierwszej instalacji systemu Unix na komputer DEC PDP-7
- **1972** - język NB (New B), nazwany później C - Dennis Ritchie, Bell Laboratories, New Jersey, system Unix na komputerze DEC PDP-11
 - 90% kodu systemu Unix oraz większość programów działających pod jego kontrolą napisane w C
- **1978** - książka „The C Programming Language” (Kernighan, Ritchie), pierwszy podręcznik, nieformalna definicja standardu (**K&R**)

Język C - krótka historia (2/2)

- **1989** - standard ANSI X3.159-1989 „Programming Language C” (**ANSI C, C89**)
- **1990** - adaptacja standardu ANSI C w postaci normy ISO/IEC 9899:1990 (**C90**)
- **1999** - norma ISO/IEC 9899:1999 (**C99**)
- **2011** - norma ISO/IEC 9899:2011 (**C11**)
- **2018** - norma ISO/IEC 9899:2018 (**C18** lub **C17**)

Język C - pierwszy program

- Niesformatowany plik tekstowy o odpowiedniej składni i mający rozszerzenie `.c`
- Kod najprostszego programu:

```
#include <stdio.h>

int main(void)
{
    printf("Witaj swiecie\n");
    return 0;
}
```

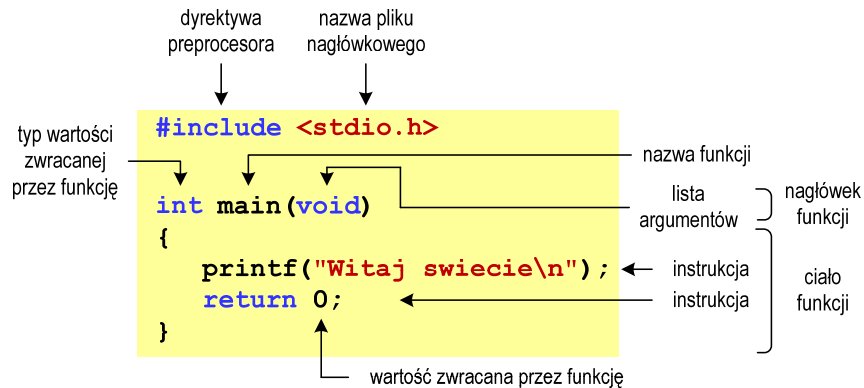
- Program konsolowy - wyświetla w konsoli tekst `Witaj swiecie`

Język C - pierwszy program

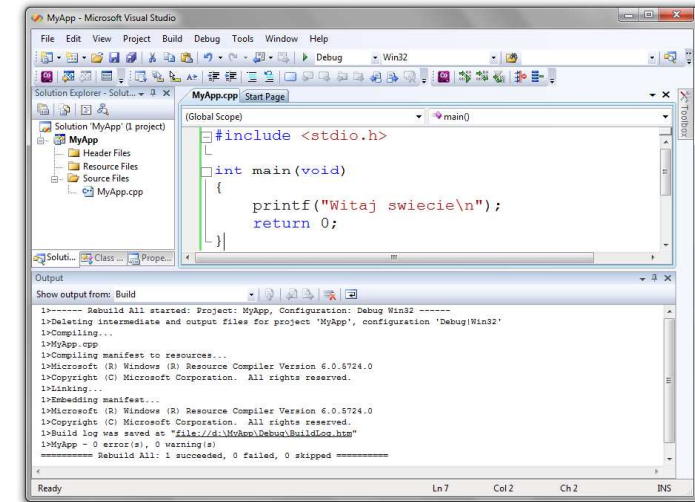
- Wynik uruchomienia programu:



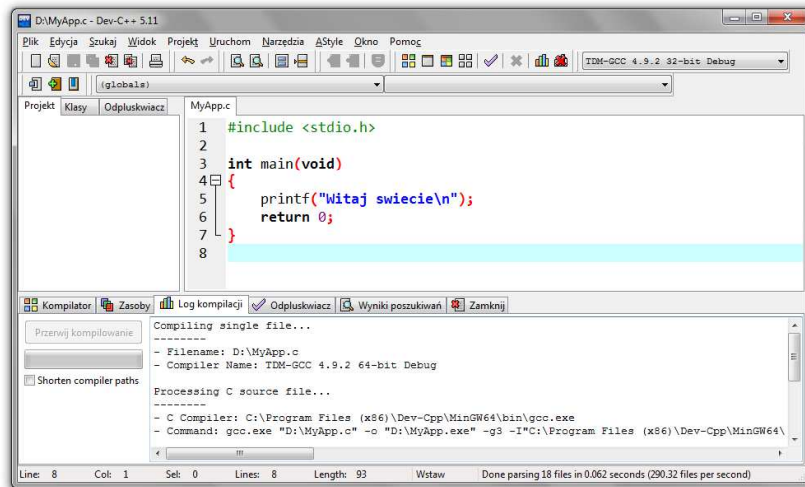
Język C - struktura programu



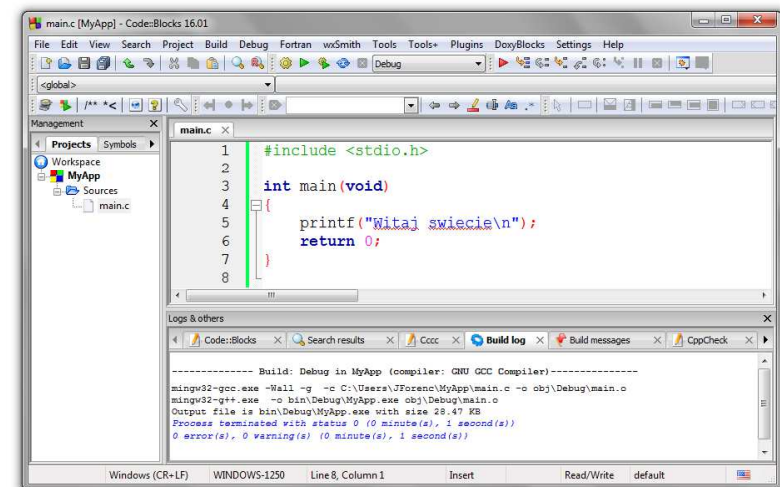
Microsoft Visual Studio 2008



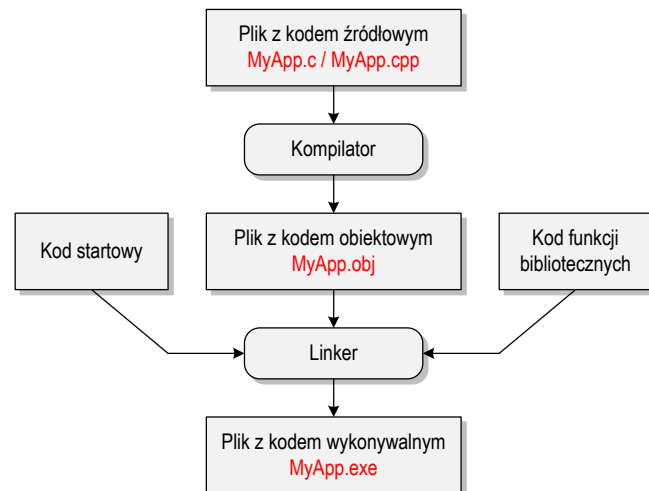
Dev-C++ 5.11



Code::Blocks 16.01



Język C - kompilacja programu



Język C - zapis kodu programu

- Sposób zapisu kodu programu wpływa tylko na jego przejrzystość, a nie na kompilację i wykonanie
- W takiej postaci program także skompiluje się:

```
#include <stdio.h>
int main(void){printf("Witaj swiecie\n");return 0;}
```

- W Microsoft Visual Studio 2008 można automatycznie sformatować kod źródłowy programu - **Ctrl + K + F**
- Język C rozróżnia wielkość liter - poniższy kod nie skompiluje się:

```
#include <stdio.h>
int Main(void){printf("Witaj swiecie\n");return 0;}
```

Język C - wyświetlanie tekstu (printf)

- Znak przejścia do nowego wiersza `\n` może pojawić w dowolnym miejscu łańcucha znaków

```
printf("Witaj swiecie\n");
```

```
Witaj swiecie
```

```
printf("Witaj\nswiecie\n");
```

```
Witaj  
swiecie
```

```
printf("Witaj ");  
printf("swiecie");  
printf("\n");
```

```
Witaj swiecie
```

Język C - wyświetlenie znaków specjalnych

- Niektóre znaki pełnią specjalną funkcję i nie można wyświetlić ich w tradycyjny sposób

Opis znaku	Znak	Zapis w printf()
Cudzysłów	"	<code>\"</code>
Apostrof	'	<code>\'</code>
Ukośnik (ang. backslash)	\	<code>\\</code>
Procent	%	<code>%%</code>

```
Sciezka dostepu: "C:\dane\plik.txt"
```

```
printf("Sciezka dostepu: \"C:\\dane\\plik.txt\\\"n");
```

Język C - sekwencje sterujące

- Istnieją także inne sekwencje sterujące (ang. escape sequence)

Opis znaku	Zapis w printf()
Alarm (ang. alert), głośniczek wydaje dźwięk	<code>\a</code>
Backspace	<code>\b</code>
Wysunięcie strony (ang. form feed)	<code>\f</code>
Przejście do nowego wiersza (ang. new line)	<code>\n</code>
CR - Carriage Return (powrót na początek wiersza)	<code>\r</code>
Tabulacja pozioma (odstęp) (ang. horizontal tab)	<code>\t</code>
Tabulacja pionowa (ang. vertical tab)	<code>\v</code>

Język C - wyświetlenie znaku o podanym kodzie

- Można wyświetlić dowolny znak podając jego kod w systemie ósemkowym lub szesnastkowym

Znaczenie	Zapis
Znak o podanym kodzie ASCII (system ósemkowy)	<code>\0oo</code>
Znak o podanym kodzie ASCII (system szesnastkowy)	<code>\xhh</code>

```
printf("\127\151\164\141\152\040");  
printf("\x73\x77\x69\x65\x63\x69\x65\x21\x0A");
```

```
Witaj swiecie!
```

Język C - wyświetlenie tekstu

```
#include <stdio.h>

int main(void)
{
    printf("-----\n");
    printf(" | Punkty | Ocena |\n");
    printf("-----\n");
    printf(" | 91-100 | 5,0 |\n");
    printf(" | 81-90  | 4,5 |\n");
    printf(" | 71-80  | 4,0 |\n");
    printf(" | 61-70  | 3,5 |\n");
    printf(" | 51-60  | 3,0 |\n");
    printf(" | 0-50   | 2,0 |\n");
    printf("-----\n");

    return 0;
}
```

Punkty	Ocena
91-100	5,0
81-90	4,5
71-80	4,0
61-70	3,5
51-60	3,0
0-50	2,0

Język C - komentarze

- Komentarze są pomijane podczas kompilacji

```
/*
Nazwa: MyApp.cpp
Autor: Jarosław Forenc, Politechnika Białostocka
Data: 14-10-2019 12:15
Opis: Program wyświetlający tekst "Witaj świecie"
*/

#include <stdio.h> // zawiera deklarację printf()

int main(void) // nagłówek funkcji main()
{
    printf/* funkcja */("Witaj świecie\n");

    return 0;
}
```

Koniec wykładu nr 1

Dziękuję za uwagę!