

Informatyka 1

Politechnika Białostocka - Wydział Elektryczny
Elektrotechnika, semestr II, studia stacjonarne I stopnia
Rok akademicki 2018/2019

Wykład nr 1 (01.03.2019)

dr inż. Jarosław Forenc

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:
 - poniedziałek, godz. 08:30-10:30, WE-204
 - środa, godz. 09:00-10:00, WE-204
 - piątek, godz. 15:20-16:50, WE-204 (studia zaoczne)

Program wykładu (1/2)

1. Informacja analogowa i cyfrowa. Pozycyjne i niepozycyjne systemy liczbowe. Konwersje pomiędzy systemami liczbowymi.
2. Jednostki informacji cyfrowej. Kodowanie informacji. Kodowanie znaków.
3. Kodowanie liczb. Reprezentacja liczb w systemach komputerowych: stałoprzecinkowa i zmiennoprzecinkowa. Standard IEEE 754.
4. Programowanie w języku C. Deklaracje i typy zmiennych, operatory i wyrażenia arytmetyczne, operacje wejścia-wyjścia, operatory relacyjne i logiczne, wyrażenia logiczne, instrukcja warunkowa if, instrukcja switch, operator warunkowy, pętle (for, while, do .. while), tablice jednowymiarowe.

Program wykładu (2/2)

5. Architektura komputerów. Klasyfikacja systemów komputerowych (taksonomia Flynna). Architektura von Neumana i architektura harwardzka.
6. Budowa i zasada działania komputera. Procesor, pamięć wewnętrzna i zewnętrzna. Komunikacja z urządzeniami zewnętrznymi, interfejsy komputerowe.
7. Algorytmy. Definicja algorytmu. Klasyfikacje i sposoby przedstawiania algorytmów. Rekurencja. Złożoność obliczeniowa. Sortowanie. Klasyfikacje algorytmów sortowania. Wybrane algorytmy sortowania.
8. Zaliczenie wykładu.

Literatura (1/2)

1. R. Kawa, J. Lembas: „Wykłady z informatyki. Wstęp do informatyki”. PWN, Warszawa, 2017.
2. W. Kwiatkowski: „Wprowadzenie do kodowania”. BEL Studio, Warszawa, 2010.
3. S. Gryś: „Arytmetyka komputerów w praktyce”. PWN, Warszawa, 2013.
4. W. Stallings: „Organizacja i architektura systemu komputerowego. Projektowanie systemu a jego wydajność”. WNT, Warszawa, 2004.
5. A.S. Tanenbaum: „Strukturalna organizacja systemów komputerowych”. Helion, Gliwice, 2006.
6. K. Wojtuszkiewicz: „Urządzenia techniki komputerowej. Część 1. Jak działa komputer? Część 2. Urządzenia peryferyjne i interfejsy”. PWN, Warszawa, 2011.

Literatura (2/2)

7. K. Banasiak: „Algorytmizacja i programowanie w Matlabie”. Wydawnictwo BTC, Legionowo, 2017.
8. P. Wróblewski: „Algorytmy, struktury danych i techniki programowania. Wydanie V”. Helion, Gliwice, 2015.
9. M. Sysło: „Algorytmy”. Helion, Gliwice, 2016.
10. B. Buczek: „Algorytmy. Ćwiczenia”. Helion, Gliwice, 2008.
11. G. Coldwin: „Zrozumieć programowanie”. PWN, Warszawa, 2015.
12. S. Prata: „Język C. Szkoła programowania. Wydanie VI”. Helion, Gliwice, 2016.

Efekty kształcenia i system ich oceniania

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

EK1	identyfikuje i opisuje zasadę działania podstawowych elementów systemu komputerowego
EK2	formułuje algorytmy komputerowe rozwiązujące typowe zadania inżynierskie występujące w elektrotechnice

Zaliczenie wykładu - efekty kształcenia (EK1)

- Student, który zaliczył przedmiot:
 - identyfikuje i opisuje zasadę działania podstawowych elementów systemu komputerowego
- Student, który zalicza na ocenę **dostateczny (3)**:
 - wymienia podstawowe elementy systemu komputerowego i podaje ich przeznaczenie
 - krótko charakteryzuje klasyfikację Flynna systemów komputerowych
 - wyjaśnia podstawowe pojęcia związane z architekturą i zasadą działania systemów komputerowych
 - dokonuje konwersji liczby całkowitej bez znaku z systemu dziesiętnego na system o dowolnej podstawie i z systemu o dowolnej podstawie na system dziesiętny



Zaliczenie wykładu - efekty kształcenia (EK1)

- Student, który zalicza na ocenę **dostateczny (3)** (c.d.):
 - wyjaśnia na czym polega zapis zmiennoprzecinkowy liczby rzeczywistej oraz postać znormalizowana tego zapisu
- 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
 - dokonuje konwersji liczby całkowitej ze znakiem na wybrany kod (ZM, U1, U2) i odwrotnie
 - charakteryzuje wybrane kody liczbowe (NKB, BCD, Graya) i alfanumeryczne (ASCII, ISO-8859, Unicode)

Zaliczenie wykładu - efekty kształcenia (EK1)

- Student, który zalicza na ocenę **bardzo dobry (5)** (oprócz wymagań na ocenę 4):
 - przedstawia cel stosowania oraz zasadę działania pamięci podręcznej
 - omawia sposób kodowania wartości specjalnych w standardzie IEEE 754

Zaliczenie wykładu - efekty kształcenia (EK2)

- 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 kształcenia (EK2)

- 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
 - przedstawia sposób sortowania wektora liczb stosując metodę sortowania szybkiego (Quick-Sort)

Zaliczenie wykładu

- Sprawdzian pisemny na koniec semestru:
 - sprawdzian: 24.06.2019 (poniedziałek), godz. 08:15, WE-Aula II
 - poprawa: termin do ustalenia (sesja egzaminacyjna)
- Na zaliczeniu oceniane będą dwa efekt kształcenia (EK1, EK2)
- Za każdy efekt kształcenia 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

Plan wykładu nr 1

- Język C
 - historia, struktura programu
 - kompilacja, zapis kodu
 - sekwencje sterujące, komentarze

Zaliczenie wykładu

- Ocena końcowa wyznaczana jest na podstawie sumy otrzymanych punktów za efekty kształcenia EK1 i EK2:

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

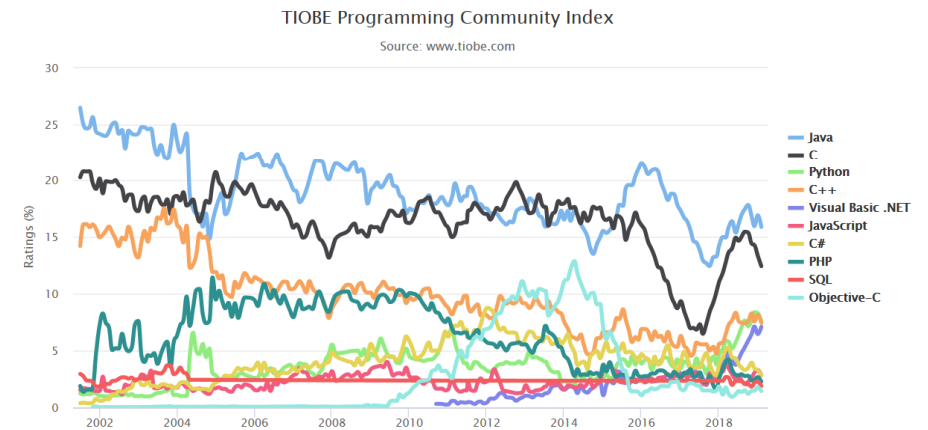
Język C - krótka historia (1/2)

- 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 - TIOBE Programming Community Index



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;
}
```

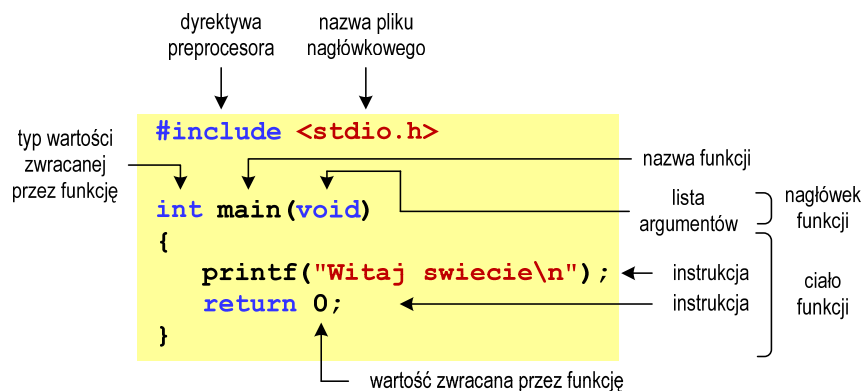
- Program konsolowy - wyświetla w konsoli tekst `Witaj swiecie`

Język C - pierwszy program

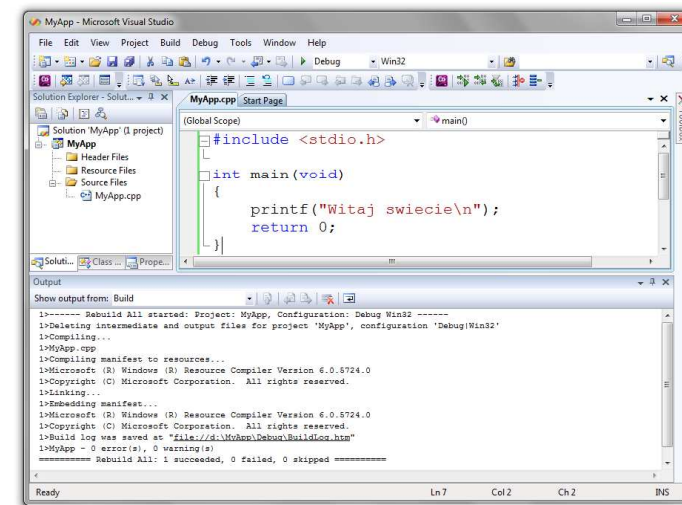
- Wynik uruchomienia programu:



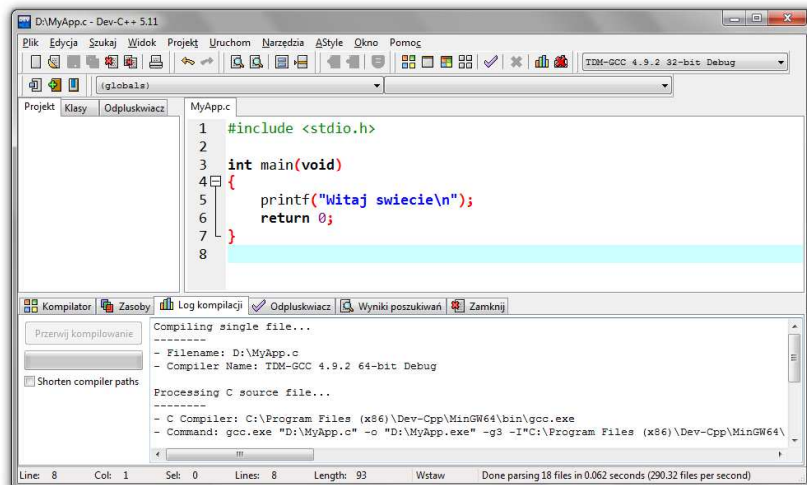
Język C - struktura programu



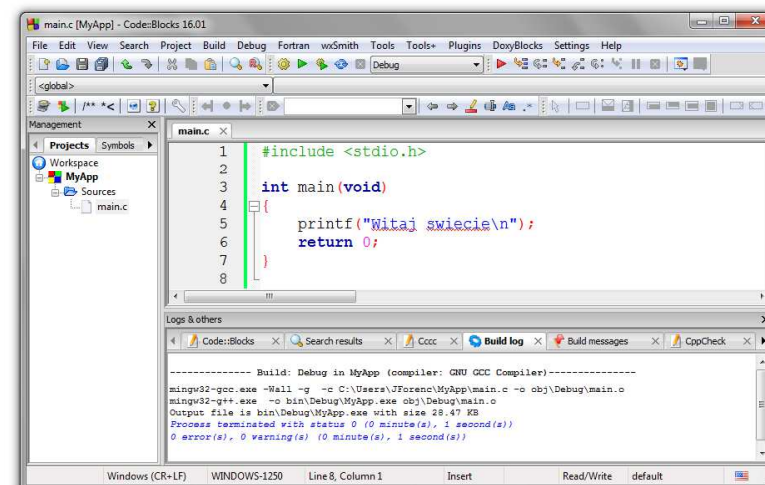
Microsoft Visual Studio 2008



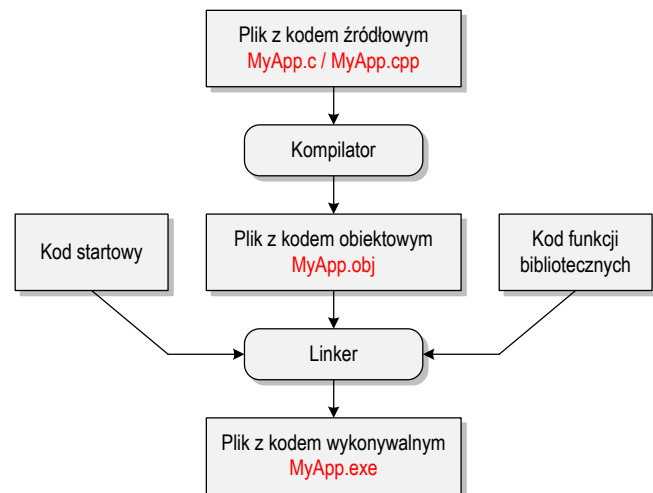
Dev-C++ 5.11



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 - 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 znaków specjalnych

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

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

Sciezka dostępu: "C:\dane\plik.txt"

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

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)	\0oo
Znak o podanym kodzie ASCII (system szesnastkowy)	\xhh

```
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: 19-02-2018 12:15  
Opis: Program wyświetlający tekst "Witaj swiecie"  
*/  
  
#include <stdio.h> // zawiera deklarację printf()  
  
int main(void) // nagłówek funkcji main()  
{  
    printf/* funkcja */("Witaj swiecie\n");  
  
    return 0;  
}
```


Koniec wykładu nr 1

Dziękuję za uwagę!