

# Informatyka 1

---

Politechnika Białostocka - Wydział Elektryczny  
Elektrotechnika, semestr II, studia stacjonarne I stopnia  
Rok akademicki 2017/2018

**Wykład nr 1 (19.02.2018)**

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](mailto:j.forenc@pb.edu.pl)
- tel. (0-85) 746-93-97
- <http://we.pb.edu.pl/~jforenc>
  - Dydaktyka - slajdy prezentowane na wykładzie
- konsultacje:
  - poniedziałek, godz. 08:30-10:00, WE-204
  - wtorek, godz. 14:00-15:00, WE-204
  - środa, godz. 12:00-14:00, WE-204
  - piątek, godz. 17:50-19:20, 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. B. Pochopień: „Arytmetyka systemów cyfrowych”. Wydawnictwo Politechniki Śląskiej, Gliwice, 2003.
2. S. Gryś: „Arytmetyka komputerów w praktyce”. PWN, Warszawa, 2013.
3. W. Stallings: „Organizacja i architektura systemu komputerowego. Projektowanie systemu a jego wydajność”. WNT, Warszawa, 2004.
4. A.S. Tanenbaum: „Strukturalna organizacja systemów komputerowych”. Helion, Gliwice, 2006.
5. K. Wojtuszkiewicz: „Urządzenia techniki komputerowej. Część 1. Jak działa komputer? Część 2. Urządzenia peryferyjne i interfejsy”. PWN, Warszawa, 2011.

## Literatura (2/2)

6. W. Malina, M. Szwoch: „Metodologia i techniki programowania”. PWN, Warszawa, 2008.
7. P. Wróblewski: „Algorytmy, struktury danych i techniki programowania. Wydanie V”. Helion, Gliwice, 2015.
8. M. Sysło: „Algorytmy”. Helion, Gliwice, 2016.
9. B. Buczek: „Algorytmy. Ćwiczenia”. Helion, Gliwice, 2008.
10. G. Coldwin: „Zrozumieć programowanie”. PWN, Warszawa, 2015.
11. S. Prata: „Język C. Szkoła programowania. Wydanie VI”. Helion, Gliwice, 2016.

## Terminy zajęć i zaliczeń

- Wykład nr 1 - 19.02.2018
- Wykład nr 2 - 05.03.2018
- Wykład nr 3 - 19.03.2018
- Wykład nr 4 - 23.04.2018
- Wykład nr 5 - 14.05.2018
- Wykład nr 6 - 28.05.2018
- Wykład nr 7 - 11.06.2018
- Wykład nr 8 - 18.06.2018 (1 h, 12:15-13:00)
  
- Zaliczenie - 11.06.2018 (poniedziałek), godz. 12:15, WE-030
- Zaliczenie poprawkowe - sesja egzaminacyjna

## Zaliczenie wykładu - efekty kształcenia

- EK1

identyfikuje i opisuje zasadę działania podstawowych elementów systemu komputerowego

- EK2

formułuje algorytmy komputerowe rozwiązujące typowe zadania inżynierskie występujące w elektrotechnice

- Szczegółowe zasady zaliczania znajdują się na stronie internetowej (<http://we.pb.edu.pl/~jforenc>) oraz w systemie USOS



## Zaliczenie wykładu

- 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

- Ocena końcowa jest średnią arytmetyczną otrzymanych ocen:

Średnia	Ocena	Średnia	Ocena
4,75 - 5,00	5,0	3,25 - 3,74	3,5
4,25 - 4,74	4,5	3,00 - 3,24	3,0
3,75 - 4,24	4,0	0 - 2,99	2,0

# Plan wykładu nr 1

- Podstawowe pojęcia: informatyka i informacja
- Informacja analogowa i cyfrowa
- Systemy liczbowe
  - liczby i cyfry
  - systemy pozycyjne (dziesiętny, dwójkowy, szesnastkowy)
  - systemy niepozycyjne (rzymski)
- Jednostki informacji cyfrowej
  - bit, bajt, słowo, FLOPS
- Język C
  - historia, struktura programu
  - kompilacja, zapis kodu
  - sekwencje sterujące, komentarze

# Informatyka

- **Informatyka** (ang. computer science)
  - dziedzina nauki i techniki zajmująca się gromadzeniem, przetwarzaniem i wykorzystywaniem **informacji**
  - w języku polskim termin informatyka zaproponował w październiku 1968 r. prof. Romuald Marczyński na konferencji poświęconej „maszynom matematycznym”
  - wzorem nazwy były francuskie **informatique** i niemieckie **Informatik**
  
- **Informatykę** można rozpatrywać jako:
  - samodzielną dyscyplinę naukową
  - narzędzie wykorzystywane przez inne nauki
  - gałąź techniki
  - przemysł wytwarzający sprzęt (hardware) i oprogramowanie (software)

# Informacja

- **Informatyka** (ang. computer science)
  - dziedzina nauki i techniki zajmująca się gromadzeniem, przetwarzaniem i wykorzystywaniem **informacji**
- **Informacja** - wielkość abstrakcyjna, która może być:
  - przechowywana w pewnych obiektach
  - przesyłana pomiędzy pewnymi obiektami
  - przetwarzana w pewnych obiektach
  - stosowana do sterowania pewnymi obiektami
- **Dane** - surowe fakty i liczby
- **Przetwarzanie danych** - logicznie powiązany zespół czynności pozwalających na uzyskanie z danych niezbędnych informacji



# Informacja

- Co oznaczają poniższe dane?

00010101000001110001010000010000



00010101	00000111	00010100	00010000
----------	----------	----------	----------

Kod binarny?



0001	0101	0000	0111	0001	0100	0001	0000
------	------	------	------	------	------	------	------

A może BCD?



1	5	0	7	1	4	1	0
---	---	---	---	---	---	---	---

Liczba: 15 071 410 ?



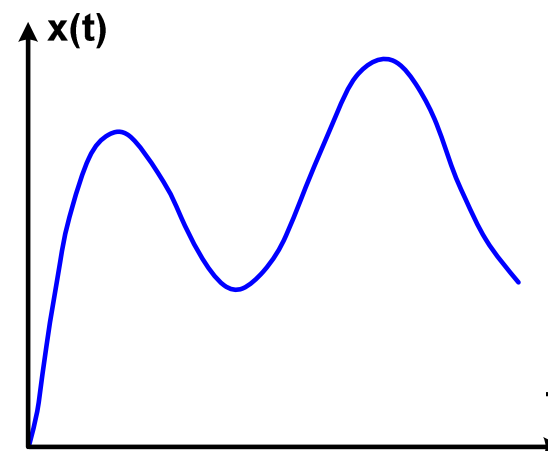
15 lipca 1410 roku

Data !!!

## Informacja analogowa i cyfrowa

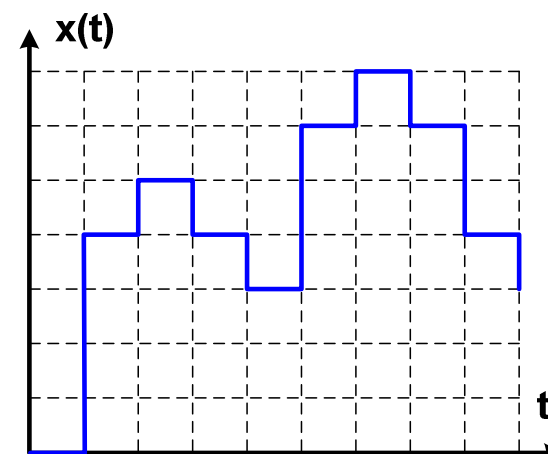
### ■ Sygnał analogowy

- może przyjmować dowolną wartość z ciągłego przedziału (nieskończonego lub ograniczonego zakresem zmienności)
- wartości mogą zostać określone w każdej chwili czasu dzięki funkcji matematycznej opisującej dany sygnał



### ■ Sygnał cyfrowy

- dziedzina i zbiór wartości są dyskretne
- sygnał ciągły, który może zmieniać swoją wartość tylko w określonych chwilach czasu i może przyjmować tylko określone wartości



# Informacja analogowa i cyfrowa

## ■ Zalety sygnałów cyfrowych:

- odporne na zakłócenia
- powtarzalne (np. kopia filmu na DVD i VHS)
- możliwość przesyłania na duże odległości
- możliwość szyfrowania sygnału (kryptografia)
- niższe koszty przetwarzania

## ■ Wady sygnałów cyfrowych:

- ograniczenie częstotliwości próbkowania (sygnał analogowy zamieniony na cyfrowy i ponownie na analogowy nie jest już tym samym sygnałem)

## Liczby i cyfry

- **Liczba** - pojęcie abstrakcyjne, abstrakcyjny wynik obliczeń, wartość
  - umożliwia wyrażenie wyniku liczenia przedmiotów oraz mierzenia wielkości
- **Cyfra** - umowny znak (symbol) stosowany do zapisu liczby
  - liczba znaków służących do zapisu jest zależna od **systemu liczbowego** i przyjętego sposobu zapisu
  - system dziesiętny - 10 znaków
  - system szesnastkowy - 16 znaków
  - system rzymski - 7 znaków
- Cyfry rzymskie

<b>I</b>	<b>V</b>	<b>X</b>	<b>L</b>	<b>C</b>	<b>D</b>	<b>M</b>
<i>1</i>	<i>5</i>	<i>10</i>	<i>50</i>	<i>100</i>	<i>500</i>	<i>1000</i>



## Liczby i cyfry

- Cyfry arabskie (pochodzą z Indii)
  - arabskie, standardowe europejskie

1	2	3	4	5	6	7	8	9	0
---	---	---	---	---	---	---	---	---	---

- indyjsko-arabskie

१	२	३	४	०	६	७	८	९	.
1	2	3	4	5	6	7	8	9	0

- wschodnio-indyjsko-arabskie

१	२	३	४	५	६	७	८	९	.
1	2	3	4	5	6	7	8	9	0

- W niektórych systemach jako cyfry stosowane są litery, np.

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

# Liczby i cyfry

- Inne przykłady zapisu cyfr i liczb:

cyfry etruskie

I	Λ	X	XX	ΛXX	↑	*	(C)	⊕	(⊕)
1	5	10	20	25	50	100	1000		

cyfry grecko-jońskie

α	β	γ	δ	ε	ς	ζ	η	θ
1	2	3	4	5	6	7	8	9
ι	κ	λ	μ	ν	ξ	ο	π	ρ
10	20	30	40	50	60	70	80	90
ϱ	σ	τ	υ	φ	χ	ψ	ω	Ͱ
100	200	300	400	500	600	700	800	900
Ϡ	β	γ	δ	ε	ς	ζ	η	θ
1000	2000	3000	4000	5000	6000	7000	8000	9000
ϠͰεβ								

cyfry w pisowni chińskiej

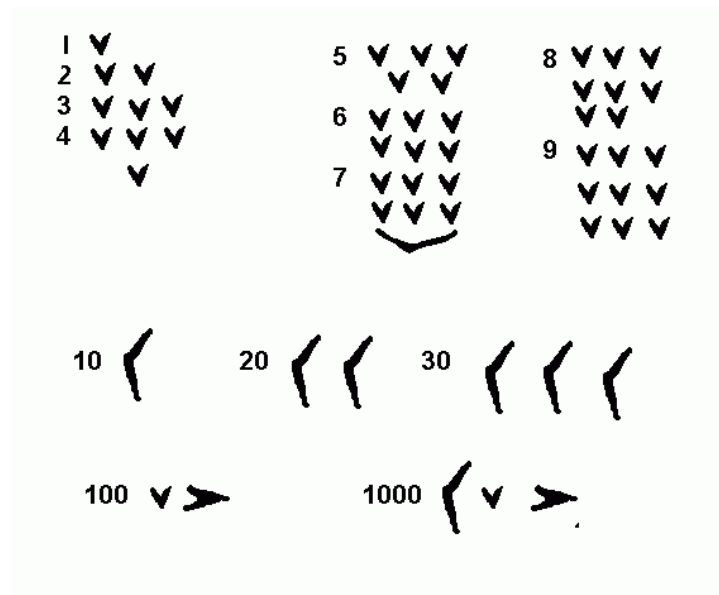
jeden	一	sześć	六
dwa	二	siedem	七
trzy	三	osiem	八
cztery	四	dziewięć	九
pięć	五	dziesięć	十
zero	另		

[uczniechinskiego.com](http://uczniechinskiego.com)

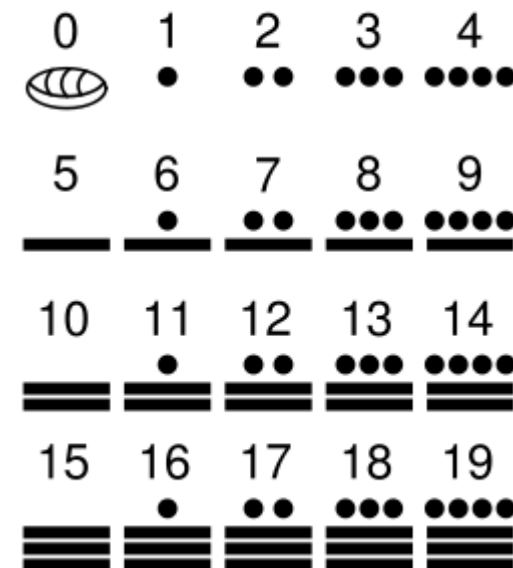
# Liczby i cyfry

- Inne przykłady zapisu cyfr i liczb:

liczby w piśmie klinowym  
(Babilończycy)

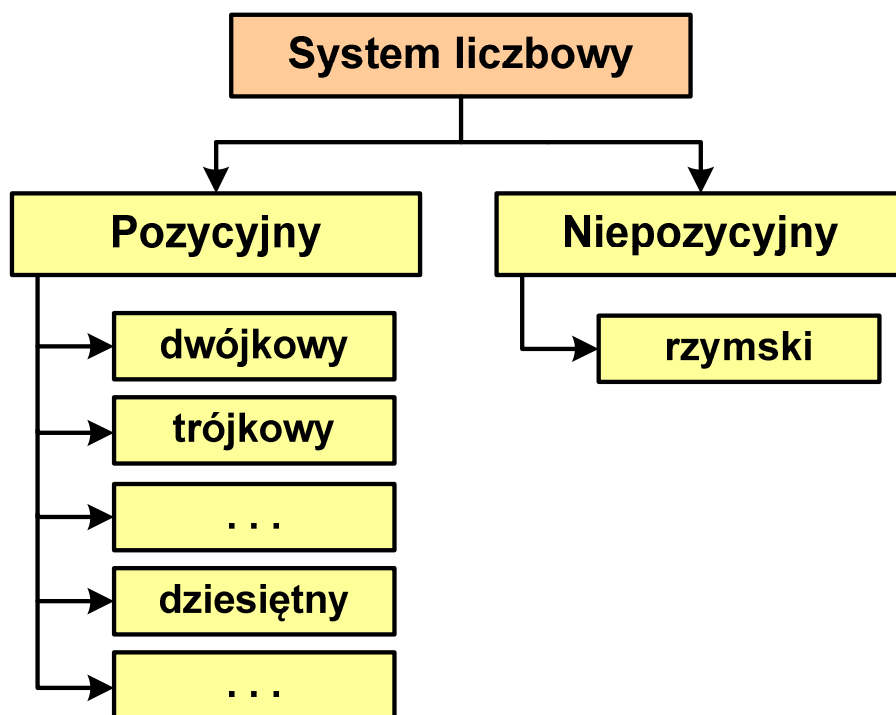


system prekolumbijski



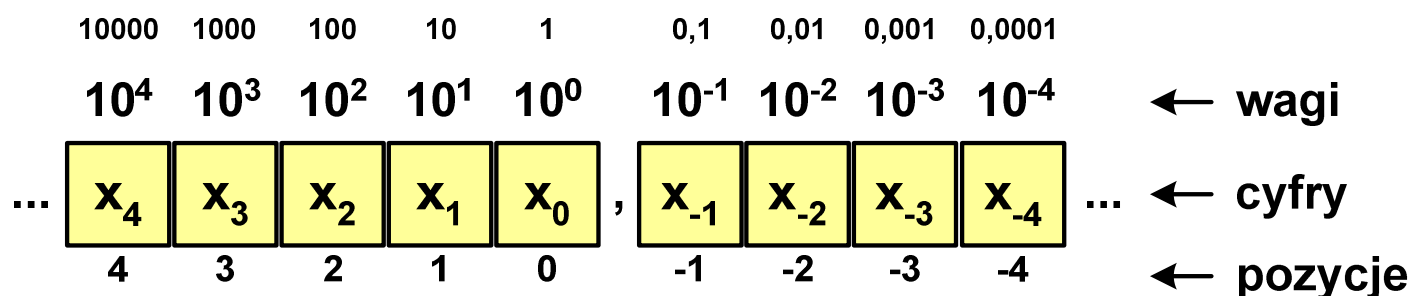
# Systemy liczbowe

- **System liczbowy** - zbiór zasad umożliwiających zapis liczb za pomocą cyfr oraz wykonywanie działań na tych liczbach



- **Pozycyjny** - znaczenie cyfry jest zależne od miejsca (pozycji), które zajmuje ona w liczbie
  - system dziesiętny - liczba **111** (każda cyfra ma inne znaczenie)
- **Niepozycyjny** - znaczenie cyfry jest niezależne od miejsca położenia w liczbie
  - system rzymski - liczba **III**

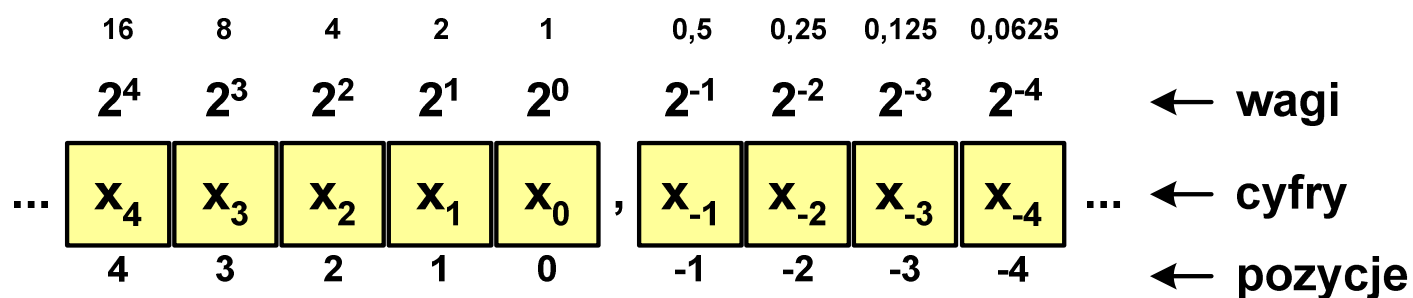
## System dziesiętny (ang. decimal)



- $p$  - podstawa systemu pozycyjnego,  $D$  - zbiór dozwolonych cyfr
- $p = 10$ ,  $D = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

$$\begin{aligned}
 & \begin{array}{cccccc}
 & 10^3 & 10^2 & 10^1 & 10^0 & 10^{-1} & 10^{-2} \\
 & \boxed{1} & \boxed{4} & \boxed{0} & \boxed{8} & , & \boxed{2} & \boxed{5} \\
 & \diagdown & \diagdown & \diagdown & \diagdown & & \diagdown & \diagdown \\
 1408,25_{(10)} = & & & & & & & \\
 = & \boxed{1 \cdot 10^3} & + & \boxed{4 \cdot 10^2} & + & \boxed{0 \cdot 10^1} & + & \boxed{8 \cdot 10^0} & + & \boxed{2 \cdot 10^{-1}} & + & \boxed{5 \cdot 10^{-2}} \\
 = & 1000 & + & 400 & + & 0 & + & 8 & + & 0,2 & + & 0,05
 \end{array}
 \end{aligned}$$

## System dwójkowy (ang. binary)

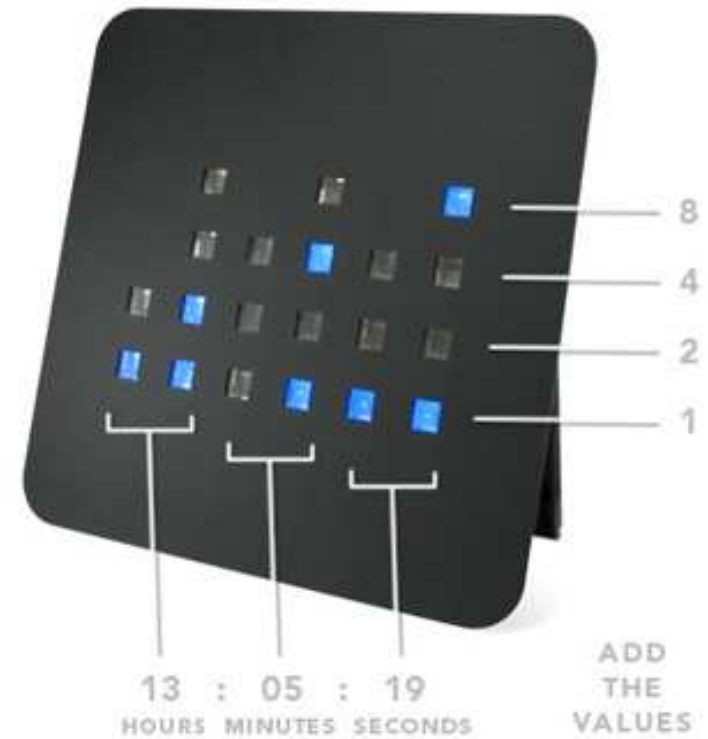


- w systemie dwójkowym:  $p = 2$ ,  $D = \{0, 1\}$

$$\begin{aligned}
 & \begin{array}{ccccccc}
 & 2^3 & 2^2 & 2^1 & 2^0 & 2^{-1} & 2^{-2} & 2^{-3} \\
 & \boxed{1} & \boxed{1} & \boxed{0} & \boxed{1} & \boxed{1} & \boxed{0} & \boxed{1} \\
 & \diagdown & \diagdown & \diagdown & \diagdown & \diagdown & \diagdown & \diagdown \\
 1101,101_{(2)} & = & \boxed{1 \cdot 2^3} & + & \boxed{1 \cdot 2^2} & + & \boxed{0 \cdot 2^1} & + & \boxed{1 \cdot 2^0} & + & \boxed{1 \cdot 2^{-1}} & + & \boxed{0 \cdot 2^{-2}} & + & \boxed{1 \cdot 2^{-3}} \\
 & = & 8 & + & 4 & + & 0 & + & 1 & + & 0,5 & + & 0 & + & 0,125 \\
 & = & 13,625_{(10)}
 \end{array}
 \end{aligned}$$

## System dwójkowy - zastosowania

- Powszechnie używany w informatyce, technice cyfrowej



## System szesnastkowy (ang. hexadecimal)

- System heksadecymalny
- $p = 16$ ,  $D = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$
- Powszechnie używany w informatyce - jeden bajt można zapisać za pomocą tylko dwóch cyfr szesnastkowych

$$3A5D_{(16)} = 3 \cdot 16^3 + 10 \cdot 16^2 + 5 \cdot 16^1 + 13 \cdot 16^0 = 14941_{(10)}$$

- Sposoby zapisu liczb w systemie szesnastkowym:

3A5Dh

0x3A5D

#3A5D

3A5D<sub>(16)</sub>

3A5D<sub>16</sub>

3A5D<sub>hex</sub>

(3A5D)<sub>hex</sub>

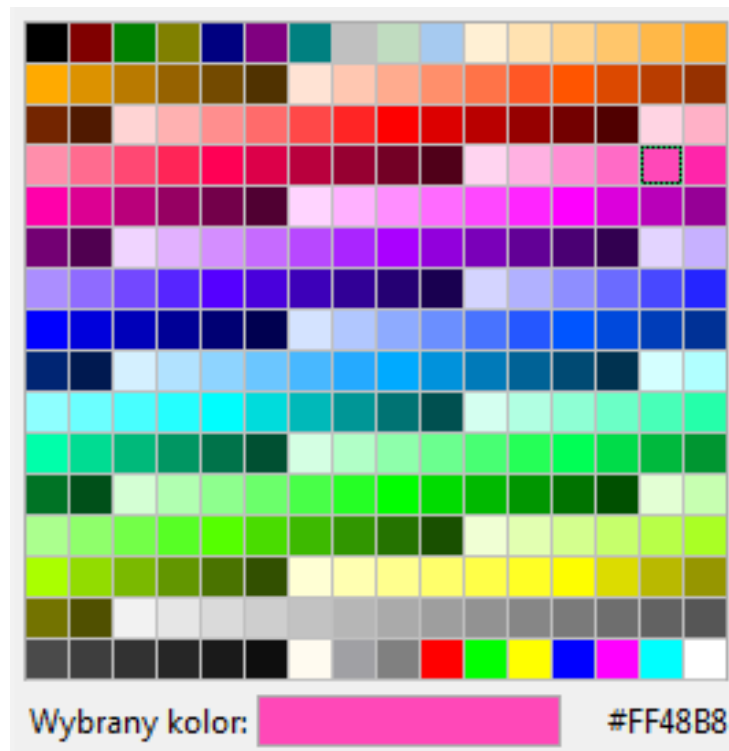
(3A5D)<sub>16</sub>

\$3A5D

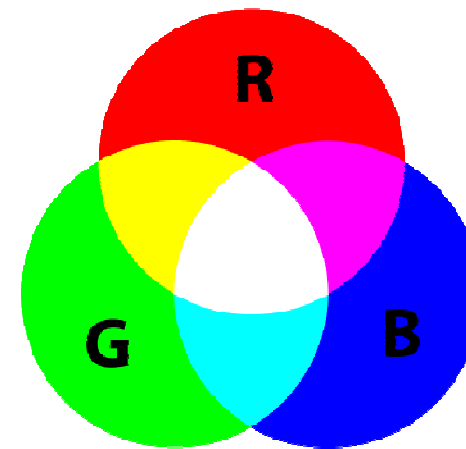


## System szesnastkowy - zastosowania

- Zapis 24-bitowego koloru RGB (Red-Green-Blue), 16 mln kolorów
- Każda barwa przyjmuje wartość z zakresu:  $0..255_{(10)}$ ,  $00..FF_{(16)}$



#FF48B8



## System szesnastkowy - zastosowania

- Zapis 24-bitowego koloru RGB (**Red-Green-Blue**), 16 mln kolorów
- Kolory w dokumentach HTML:

```
<BODY bgcolor="#336699" text="#000000" link="#FFFF00"  
vlink="#33FFFF" alink="#FF0000">
```

The screenshot shows a website interface with a dark blue background. On the left, there are two buttons: 'ARCHIWUM' and 'ENGLISH'. The main content area is titled 'Studia stacjonarne:' and lists a schedule for Monday and Tuesday. The schedule includes times and links to course pages.

<b>ARCHIWUM</b>	<b>Studia stacjonarne:</b>
<b>ENGLISH</b>	<b>Poniedziałek:</b>
	12:15 - 14:00 <a href="#">Informatyka 1 - wykład</a> , sem. 2 ED, WE-Aula II
	<b>Wtorek:</b>
	08:30 - 10:00 <a href="#">Informatyka 1 - prac.</a> , sem. 2 ED, gr. PS3, WE-110
	12:15 - 13:45 <a href="#">Informatyka 1 - prac.</a> , sem. 2 ED, gr. PS2, WE-110
	14:15 - 15:45 <a href="#">Informatyka 1 - prac.</a> , sem. 2 ED, gr. PS1, WE-110

## System szesnastkowy - zastosowania

- 48-bitowy adres fizyczny urządzenia (MAC - Media Access Control)

**88 : AD : D2 : 09 : 41 : 3B**

└──────────┬──────────┘ └──────────┬──────────┘

producent      numer egzemplarza

- <http://hwaddress.com>

OUI	MAC range	Company
88-AD-D2	88-AD-D2-00-00-00 - 88-AD-D2-FF-FF-FF	Samsung Electronics Co.,Ltd

## Przykład systemu niepozycyjnego - system rzymski

- W systemie rzymskim posługujemy się siedmioma znakami:  
**I** - 1    **V** - 5    **X** - 10    **L** - 50    **C** - 100    **D** - 500    **M** - 1000
- Za pomocą dostępnych symboli można określić liczby od **1** do **3999**
- System **addytywny** - wartość liczby określa się na podstawie sumy wartości cyfr, np.
  - **II** ( $1 + 1 = 2$ ),    **XXX** ( $10 + 10 + 10 = 30$ )
  - **CLX** ( $100 + 50 + 10 = 160$ ),    **MMXII** ( $1000 + 1000 + 10 + 1 + 1 = 2012$ )
- Wyjątkiem od powyższej zasady są liczby do opisu których używa się odejmowania, np.
  - **IV** ( $5 - 1 = 4$ ),    **IX** ( $10 - 1 = 9$ ),    **XL** ( $50 - 10 = 40$ ),    **XC** ( $100 - 10 = 90$ )
- Stosowany w łacińskiej części Europy do końca Średniowiecza
- Niewygodny w prowadzeniu nawet prostych działań arytmetycznych, brak ułamków

## Przykład systemu niepozycyjnego - system rzymski

### ■ Zasady tworzenia liczb:

- zestawiamy odpowiednie znaki od oznaczającego liczbę największą do oznaczającego liczbę najmniejszą

$$\text{XVI} = 10(\text{X}) + 5(\text{V}) + 1(\text{I}) = 16$$

- jeżeli składnik liczby, którą piszemy, jest wielokrotnością liczby nominalnej, wtedy zapisywany jest z użyciem kilku następujących po sobie znaków

$$\text{CCC} = 100(\text{C}) + 100(\text{C}) + 100(\text{C}) = 300$$

- dodatkowo należy zachować zasadę nie pisania czterech tych samych znaków po sobie, lecz napisać jeden znak wraz ze znakiem oznaczającym wartość większą o jeden rząd liczbowy

$$\text{CD} = 500(\text{D}) - 100(\text{C}) = 400$$

## Przykład systemu niepozycyjnego - system rzymski

### ■ Zasady odczytu liczb:

- cyfry jednakowe są dodawane

$$MMM = 1000(M) + 1000(M) + 1000(M) = 3000$$

- cyfry mniejsze stojące przed większymi są odejmowane od nich

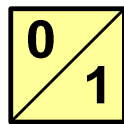
$$CDXCIV = 500(D) - 100(C) + 100(C) - 10(X) + 5(V) - 1(I) = 494$$

- cyfry mniejsze stojące za większymi są do nich dodawane

$$MDCLX = 1000(M) + 500(D) + 100(C) + 50(L) + 10(X) = 1660$$

## Jednostki informacji - bit

- **Bit** (ang. **b**inary digit) - podstawowa jednostka informacji stosowana w informatyce i telekomunikacji
- Określa najmniejszą ilość informacji potrzebną do stwierdzenia, który z dwóch możliwych stanów przyjął układ
- Bit przyjmuje jedną z dwóch wartości:
  - 0 (zero)
  - 1 (jeden)
- Bit jest tożsamy z cyfrą w systemie dwójkowym
- Oznaczenia bitów:
  - standard IEEE 1541 (2002) - mała litera „b”
  - standard IEC 60027 - „bit”



## Jednostki informacji - bit

- Wielokrotności bitów:

Przedrostki dziesiętne (układ SI)		
Nazwa	Symbol	Mnożnik
bit	b	---
kilobit	kb	$10^3 = 1000^1$
megabit	Mb	$10^6 = 1000^2$
gigabit	Gb	$10^9 = 1000^3$
terabit	Tb	$10^{12} = 1000^4$
petabit	Pb	$10^{15} = 1000^5$
eksabit	Eb	$10^{18} = 1000^6$
zettabit	Zb	$10^{21} = 1000^7$
jottabit	Yb	$10^{24} = 1000^8$

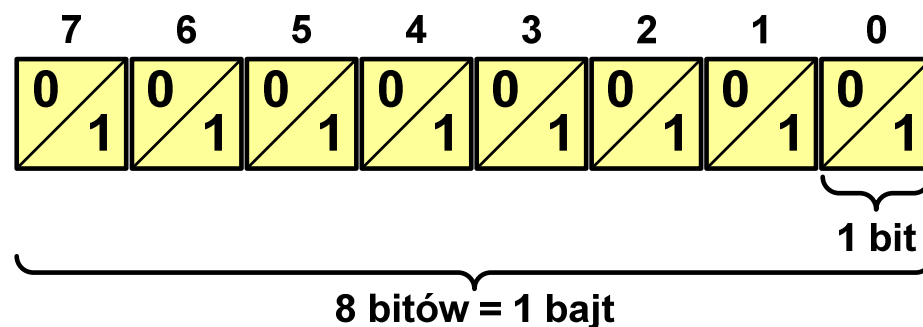
Przedrostki binarne (IEC 60027-2)		
Nazwa	Symbol	Mnożnik
bit	b	---
kibibit	Kib	$2^{10} = 1024^1$
mebibit	Mib	$2^{20} = 1024^2$
gibibit	Gib	$2^{30} = 1024^3$
tebibit	Tib	$2^{40} = 1024^4$
pebibit	Pib	$2^{50} = 1024^5$
eksbibit	Eib	$2^{60} = 1024^6$
zebibit	Zib	$2^{70} = 1024^7$
jobibit	Yib	$2^{80} = 1024^8$

- **Przedrostki binarne** - wprowadzone w 1998 roku w celu odróżnienia przedrostków o mnożniku 1000 ( $10^3$ ) od przedrostków o mnożniku 1024 ( $2^{10}$ )



## Jednostki informacji - bajt

- **Bajt** (ang. byte) - najmniejsza adresowalna jednostka informacji pamięci komputerowej składająca się z bitów
- W praktyce przyjmuje się, że jeden bajt to 8 bitów



- Za pomocą jednego bajtu można zapisać  $2^8 = 256$  różnych wartości:

0000 0000	→	0	...	→	...
0000 0001	→	1	1111 1101	→	253
0000 0010	→	2	1111 1110	→	254
...		...	1111 1111	→	255

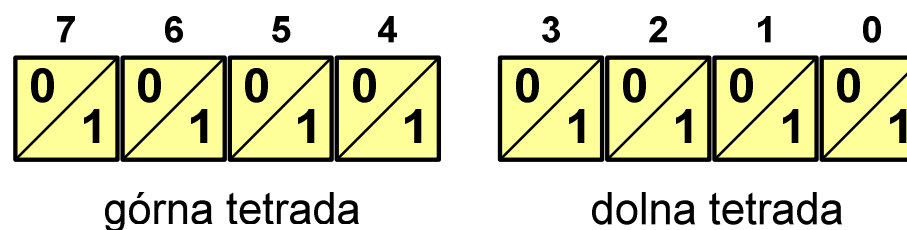
## Jednostki informacji - bajt

- W pierwszych komputerach bajt mógł mieć inną liczbę bitów: 4, 6, 7, 9, 12
- 8-bitowy bajt:
  - koniec 1956 r. - pierwsze zastosowanie
  - 1964 r. - uznanie za standard (IBM System/360)
- Inna nazwa 8-bitowego bajtu - **oktet**
- Najczęściej stosowanym skrótem dla bajtu jest wielka litera „**B**”
  - „**B**” używane jest także do oznaczania **bela** - jednostki miary wielkości ilorazowych
  - zamiast bely częściej używa się jednostki podwielokrotnej - **decybela (dB)** więc nie ma problemu z rozróżnieniem obu jednostek

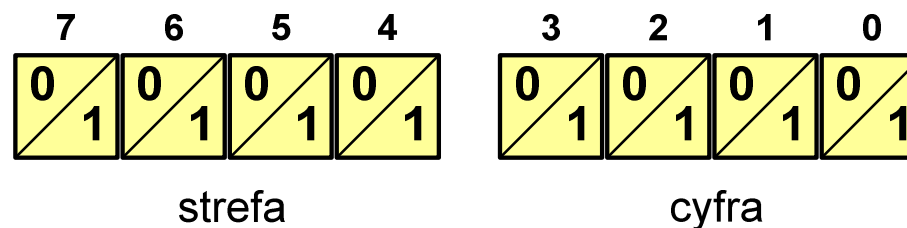


## Jednostki informacji - tetrada

- Bajt 8-bitowy można podzielić na dwie połówki 4-bitowe nazywane **tetradami** (ang. nibbles)
- Rozróżniamy bardziej znaczącą (górną) i mniej znaczącą (dolną) tetradę



- Spotyka się też określenie **strefa** i **cyfra**



## Jednostki informacji - bajt

- Wielokrotności bajtów:

Przedrostki dziesiętne (układ SI)		
Nazwa	Symbol	Mnożnik
bajt	B	---
kilobajt	kB	$10^3 = 1000^1$
megabajt	MB	$10^6 = 1000^2$
gigabajt	GB	$10^9 = 1000^3$
terabajt	TB	$10^{12} = 1000^4$
petabajt	PB	$10^{15} = 1000^5$
eksabajt	EB	$10^{18} = 1000^6$
zettabajt	ZB	$10^{21} = 1000^7$
jottabajt	YB	$10^{24} = 1000^8$

Przedrostki binarne (IEC 60027-2)		
Nazwa	Symbol	Mnożnik
bajt	B	---
kibibajt	KiB	$2^{10} = 1024^1$
mebibajt	MiB	$2^{20} = 1024^2$
gibibajt	GiB	$2^{30} = 1024^3$
tebibajt	TiB	$2^{40} = 1024^4$
pebibajt	PiB	$2^{50} = 1024^5$
eksbibajt	EiB	$2^{60} = 1024^6$
zebibajt	ZiB	$2^{70} = 1024^7$
jobibajt	YiB	$2^{80} = 1024^8$

## Jednostki informacji - bajt

- Przedrostki binarne (dwójkowe) nie zostały przyjęte przez wszystkie środowiska zajmujące się informatyką
- Producenci nośników pamięci korzystają z przedrostków dziesiętnych

Prefiks	Nazwa	System SI	System binarny	Różnica
k	kilo	$10^3 = 1000$	$2^{10} = 1024$	2,40%
M	mega	$10^6 = 1000000$	$2^{20} = 1048576$	4,86%
G	giga	$10^9 = 1000000000$	$2^{30} = 1073741824$	7,37%
T	tera	$10^{12} = 1000000000000$	$2^{40} = 1099511627776$	9,95%

- Z ulotki „Dysk Desktop HDD - zestawienie danych”, Seagate:
  - w przypadku oznaczania pojemności dysków, jeden gigabajt (oznaczany także jako „GB”) jest równy jednemu miliardowi bajtów, a jeden terabajt (oznaczany także jako „TB”) jest równy jednemu bilionowi bajtów

## Jednostki informacji - bajt

- Seagate ST1000DM003 (1 TB)
- Drive specification:
  - formatted capacity: 1000 GB (1 TB)
  - guaranteed sectors: 1,953,525,168
  - bytes per sector: 4096  
(4K physical emulated at 512-byte sectors)
- Pojemność dysku:
  - $1.953.525.168 \times 512 = 1.000.204.886.016$  bajtów
  - $1.000.204.886.016 / (1024) = 976.762.584$  kB
  - $1.000.204.886.016 / (1024 \times 1024) = 953.870$  MB
  - $1.000.204.886.016 / (1024 \times 1024 \times 1024) = 931,5$  GB

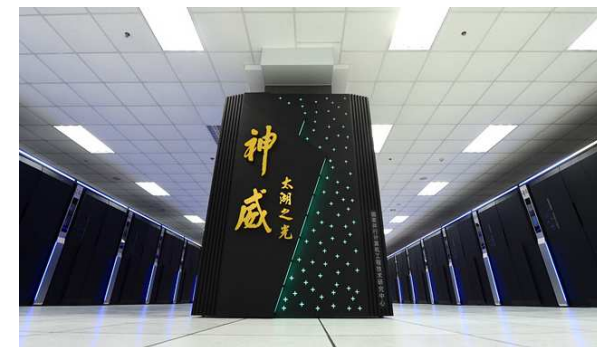


## Słowo maszynowe (słowo)

- **Słowo maszynowe** (**słowo** - ang. word) - jednostka danych używana przez określony komputer (określoną architekturę)
- Słowo składa się odgórnie określonej liczby bitów, nazywanej **długością** lub **szerokością słowa** (najczęściej jest to potęga 2, np. 8, 16, 32, 64 bity)
- Zazwyczaj wielkość słowa określa:
  - rozmiar rejestrów procesora
  - rozmiar szyny danych i szyny adresowej
- Architektury:
  - 8-bitowa: Intel 8080, Z80, Motorola 6800, Intel 8051
  - 16-bitowa: Intel 8086, Intel 80286
  - 32-bitowa: Intel od 80386 do i7, AMD od 5x86 do Athlona, ARM
  - 64-bitowa: Intel Itanium, Pentium 4/EM64T, Core 2, Core i7  
AMD Opteron, Athlon 64, Athlon II

# FLOPS

- **FLOPS** (FLoating point Operations Per Second)
  - liczba operacji zmiennoprzecinkowych na sekundę
  - jednostka wydajności układów zmiennoprzecinkowych
- Przykłady wydajności procesorów (teoretyczne):
  - Intel Core i7 975 3,46 GHz - 55,36 GFlops
  - Intel Core2 Quad Q9650 3,00 GHz - 48 GFlops
  - Intel Core2 Duo E8400 3,00 GHz - 24 GFlops
  - najszybszy system równoległy na świecie:  
Sunway TaihuLight, China - 93.014.600 GFlops  
National Supercomputing Center in Wuxi  
procesory: 40.960 szt. (260 rdzeni)  
Sunway SW26010 260C 1.45 GHz  
10.649.600 rdzeni  
[www.top500.org](http://www.top500.org)





## 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)

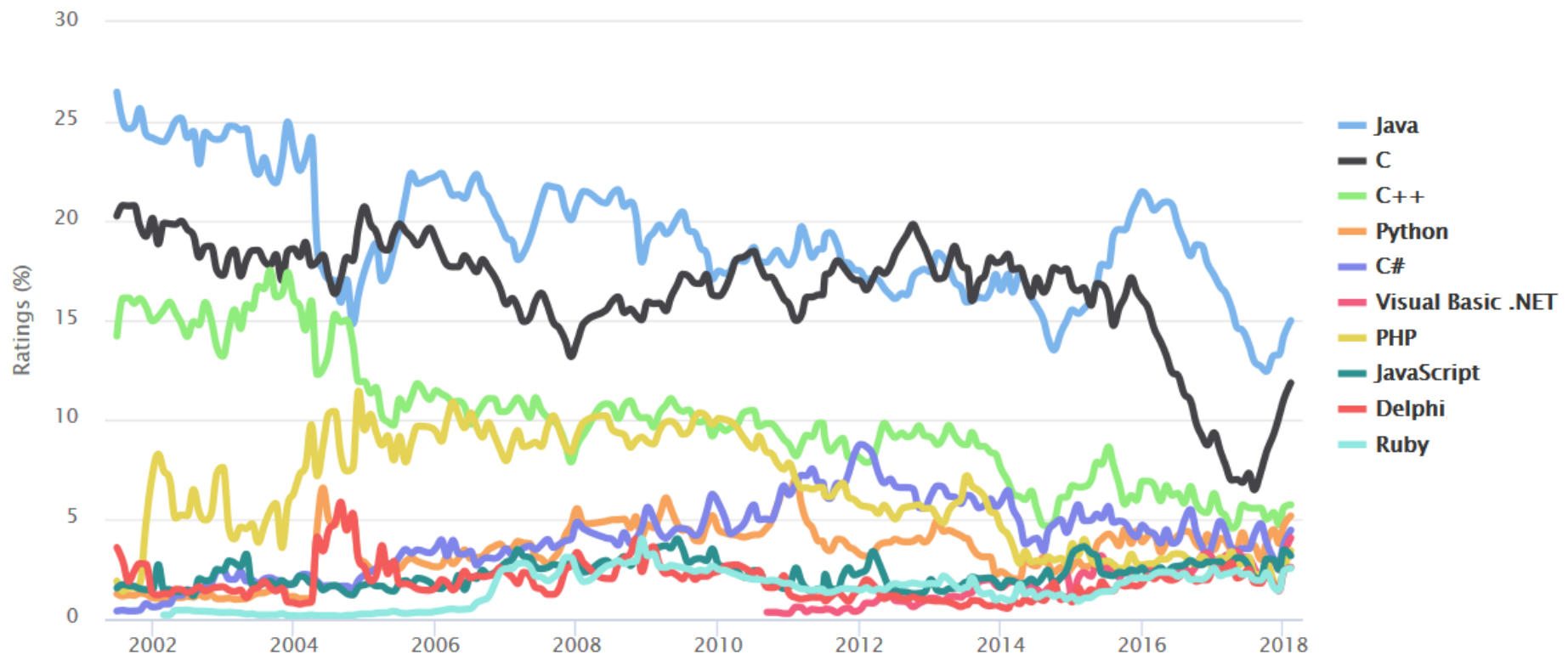
## Język C - krótka historia (2/2)

- **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)

# Język C - TIOBE Programming Community Index

TIOBE Programming Community Index

Source: [www.tiobe.com](http://www.tiobe.com)



## Język C - pierwszy program

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

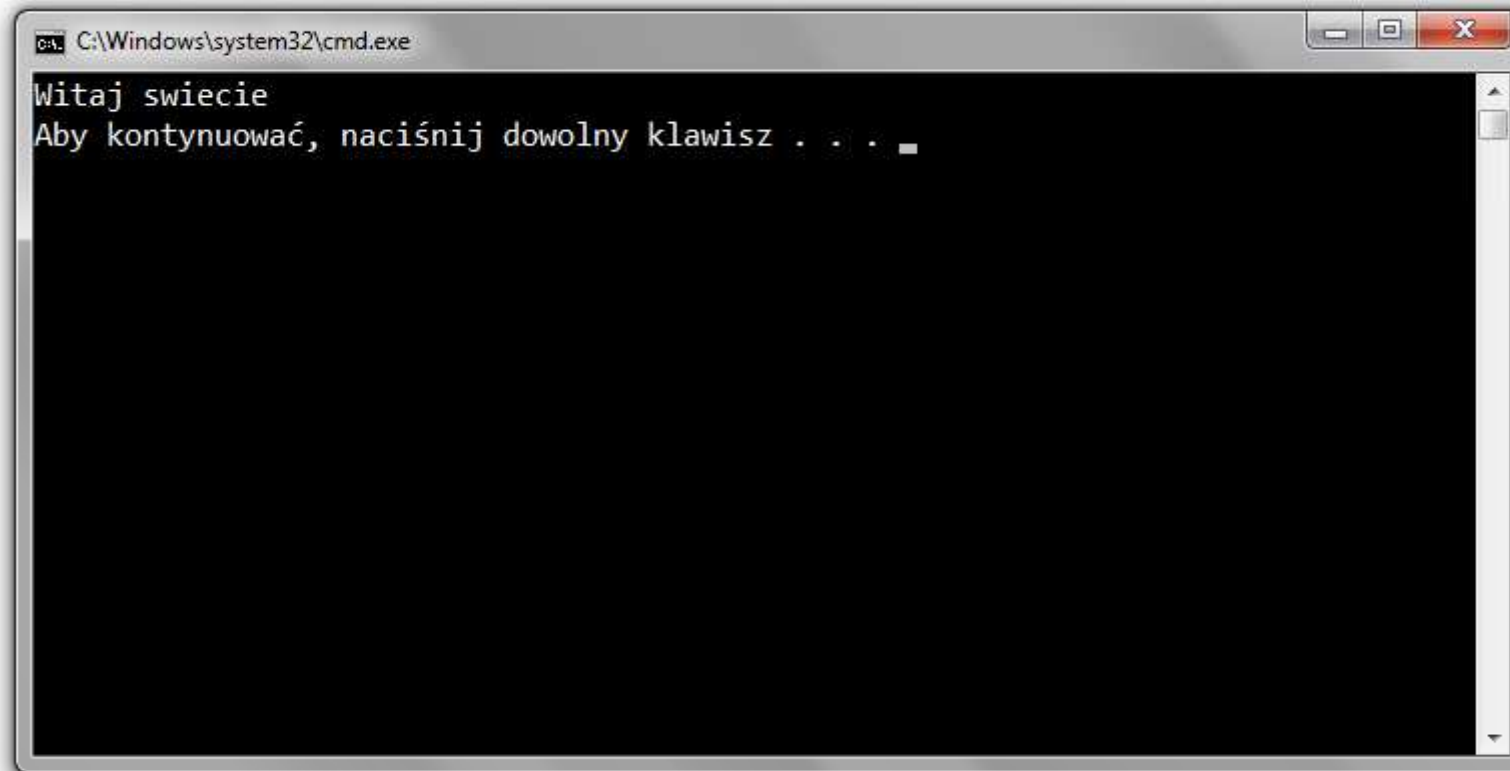
```
#include <stdio.h>

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

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

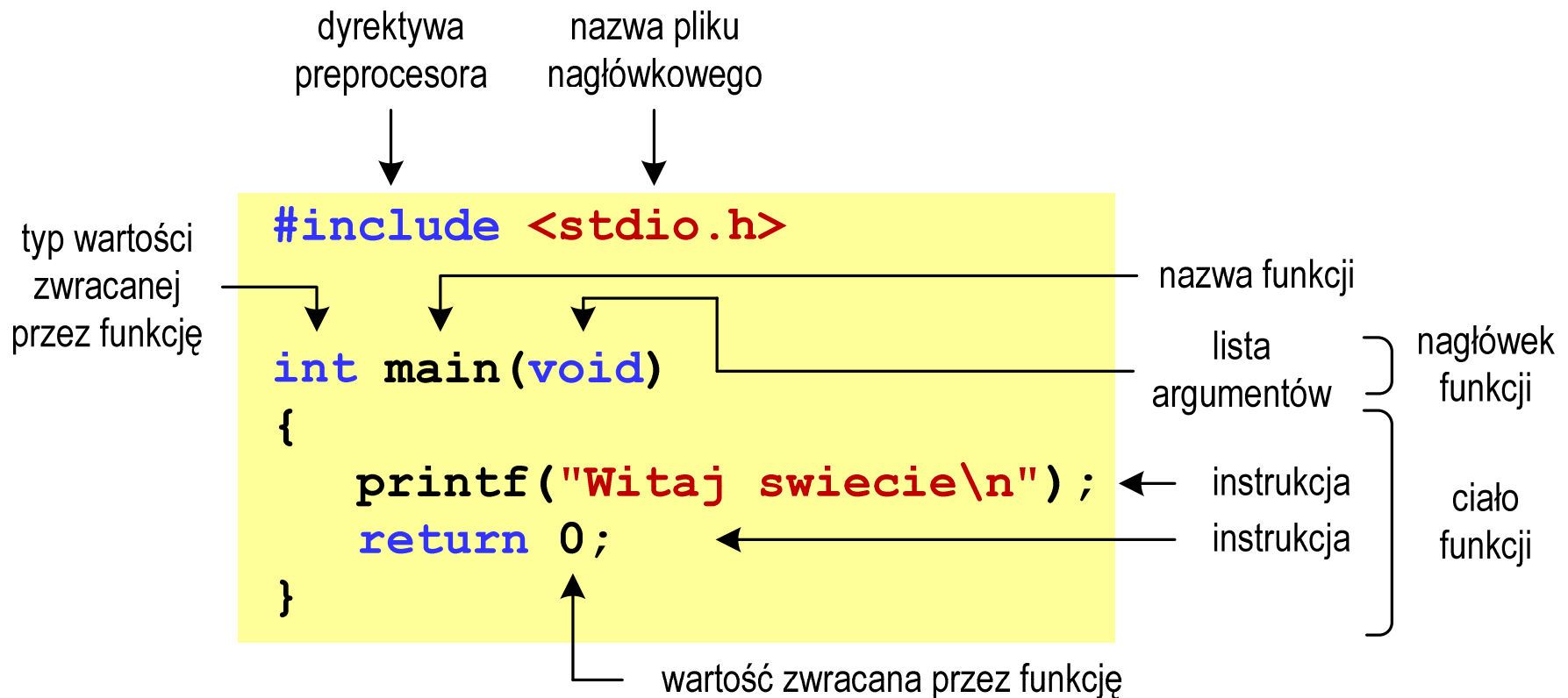
## Język C - pierwszy program

- Wynik uruchomienia programu:

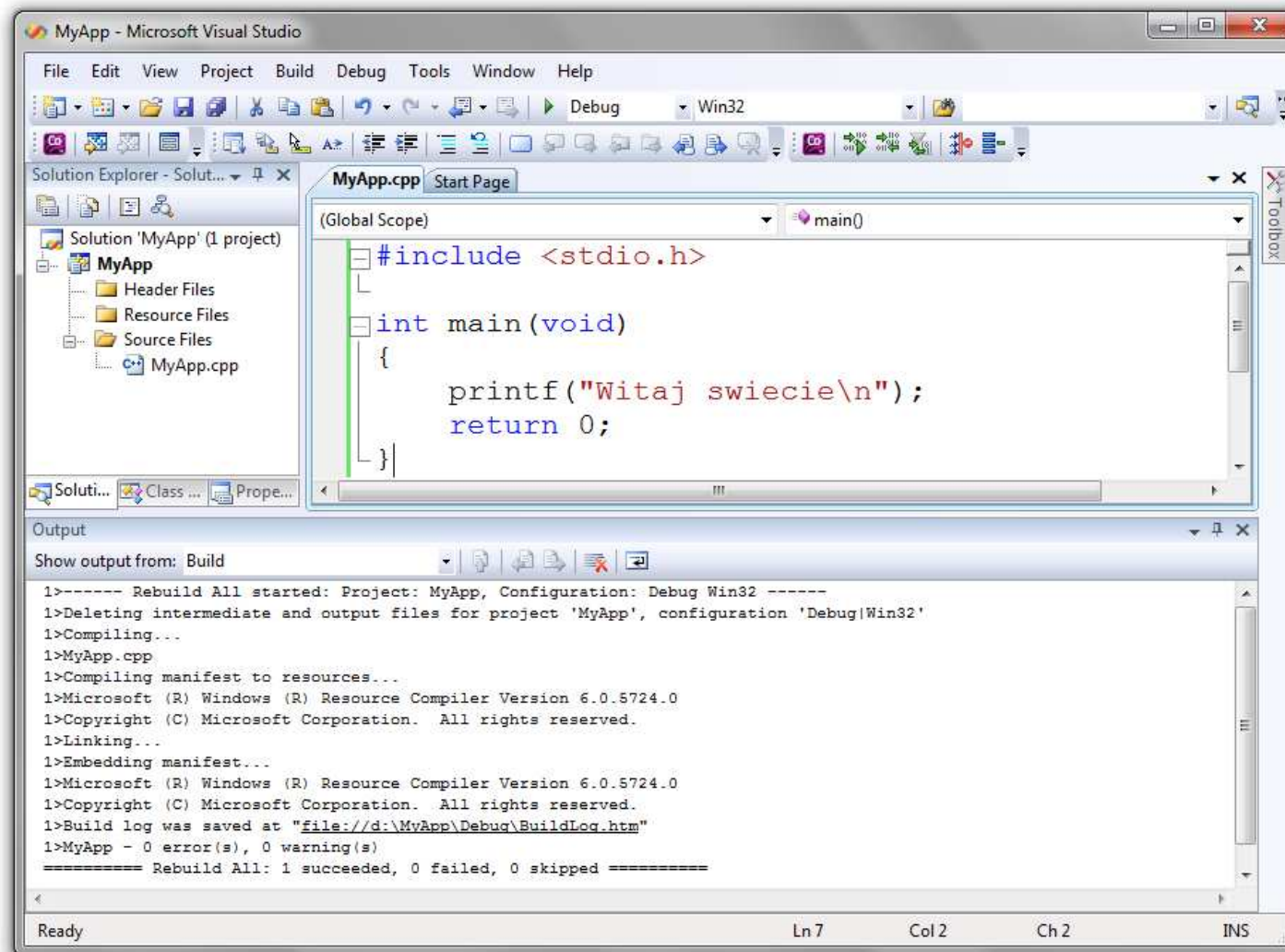


```
C:\Windows\system32\cmd.exe
Witaj swiecie
Aby kontynuować, naciśnij dowolny klawisz . . . █
```

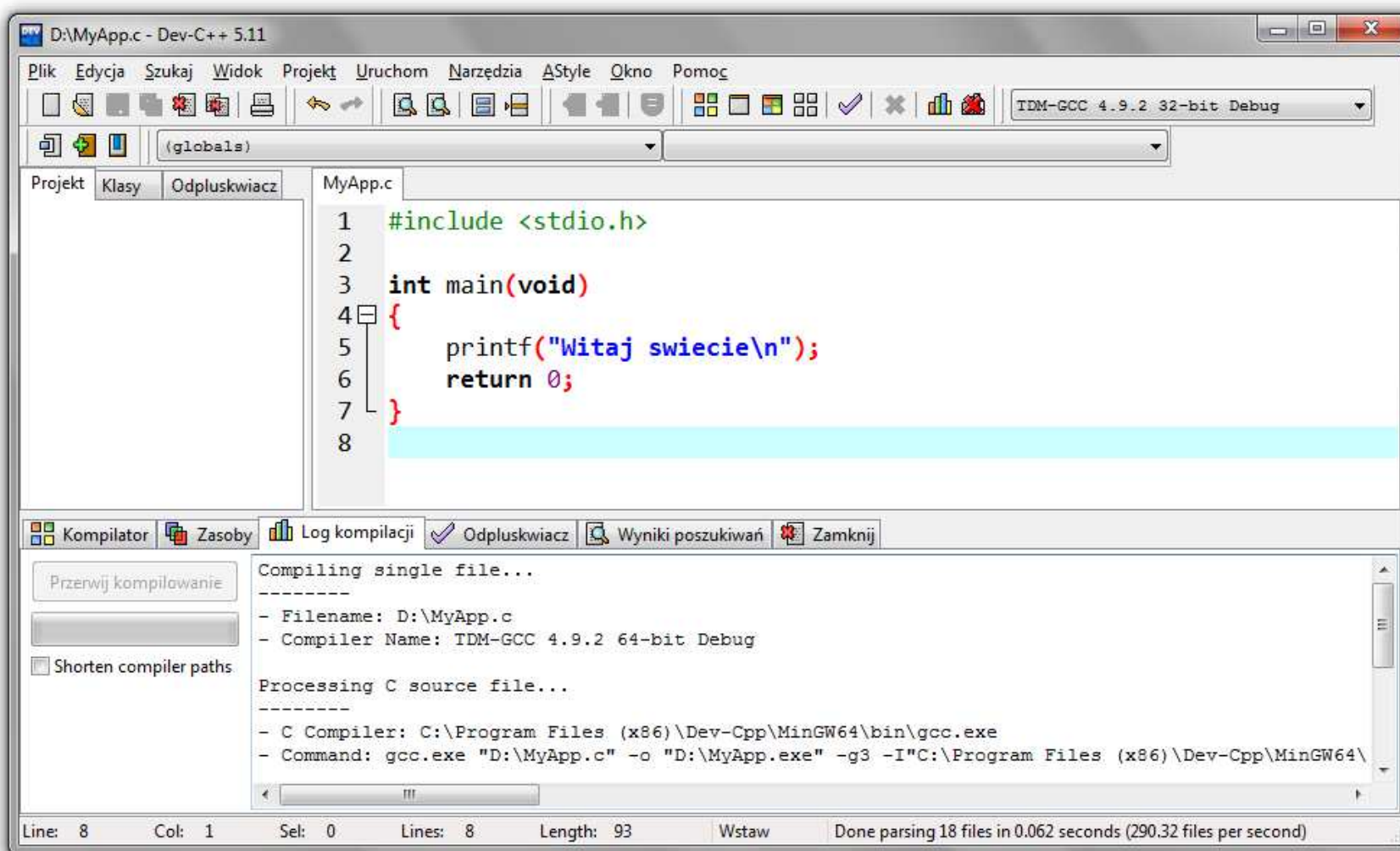
## Język C - struktura programu



# Microsoft Visual Studio 2008

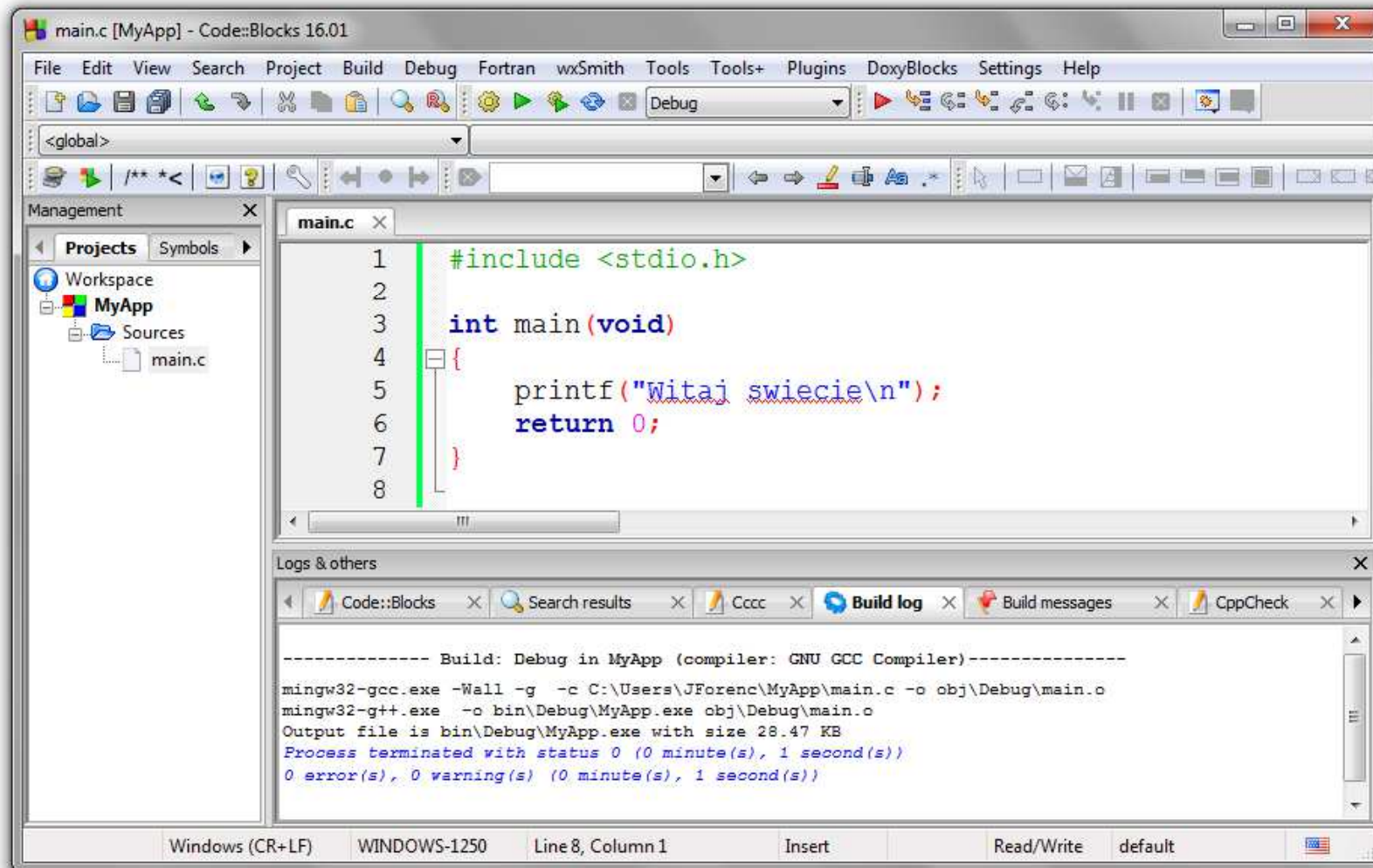


# Dev-C++ 5.11

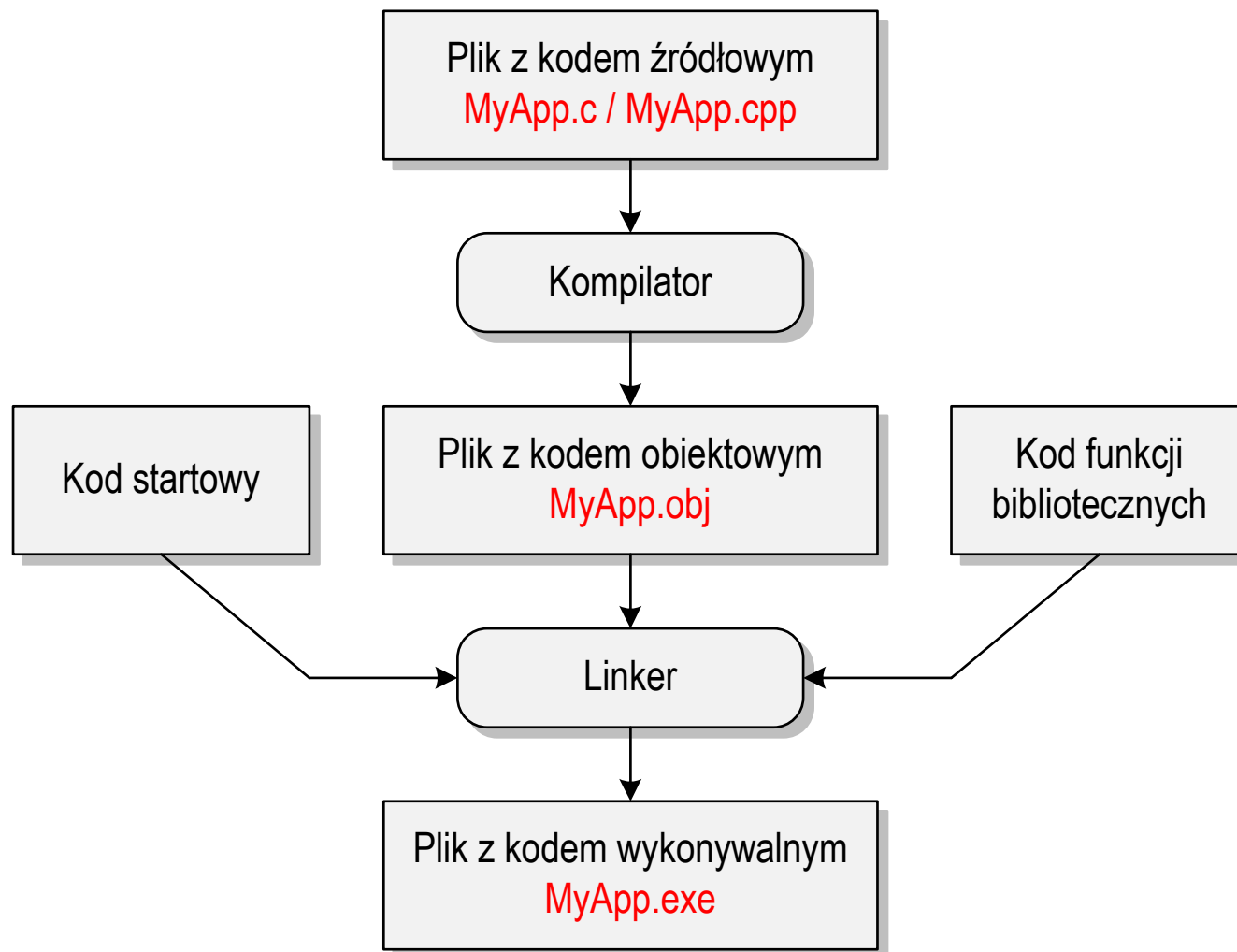




## Code::Blocks 16.01



## 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)

<b>Opis znaku</b>	<b>Zapis w printf()</b>
Alarm (ang. alert), głośniczek wydaje dźwięk	<code>\a</code>
Backspace	<code>\b</code>
Wysunięcie strony (ang. form feed)	<code>\f</code>
Przejdźcie 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 dostępu: \"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)	<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: 19-02-2018 12:15  
  Opis: Program wyświetlający tekst "Witaj świecie"  
*/  
  
#include <stdio.h>    // zawiera deklarację printf()  
  
int main(void)        // nagłówek funkcji main()  

```

Koniec wykładu nr 1

**Dziękuję za uwagę!**  
**(następny wykład: 05.03.2018)**