

Informatyka 1

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

Wykład nr 3 (15.03.2019)

dr inż. Jarosław Forenc

Plan wykładu nr 3

- Język C
 - funkcje matematyczne (math.h)
 - funkcje printf i scanf
- Pojęcia: informatyka i informacja
- Informacja analogowa i cyfrowa
- Systemy liczbowe
 - liczby i cyfry

Przykład: zamiana wzrostu w cm na stopy i cale

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

```
int main(void)
```

```
{
```

```
    float cm;        /* wzrost w cm */
```

```
    float stopy;     /* wzrost w stopach */
```

```
    float cale;      /* wzrost w calach */
```

```
    printf("Podaj wzrost w cm: ");
```

```
    scanf("%f", &cm);
```

```
    stopy = cm / 30.48f;
```

```
    cale = cm / 2.54f;
```

```
    printf("%f [cm] = %f [ft]\n", cm, stopy);
```

```
    printf("%f [cm] = %f [in]\n", cm, cale);
```

```
    return 0;
```

```
}
```

Podaj wzrost w cm: 175

175.000000 [cm] = 5.741470 [ft]

175.000000 [cm] = 68.897636 [in]

Język C - Funkcje matematyczne (math.h)

- Plik nagłówkowy **math.h** zawiera definicje wybranych stałych

Nazwa	Wartość	Znaczenie
M_PI	3.14159265358979323846	liczba pi
M_E	2.71828182845904523536	e - liczba Eulera
M_LN2	0.693147180559945309417	ln 2
M_SQRT2	1.41421356237309504880	$\sqrt{2}$

- W środowisku Visual Studio 2008 użycie stałych wymaga definicji odpowiedniej stałej (przed **#include <math.h>**)

```
#define _USE_MATH_DEFINES  
#include <math.h>
```

Język C - Funkcje matematyczne (math.h)

- Wybrane funkcje matematyczne:

Nazwa	Nagłówek	Znaczenie
abs	int abs(int x);	moduł x (x - całkowite)
fabs	double fabs(double x);	moduł x (x - rzeczywiste)
sqrt	double sqrt(double x);	pierwiastek kwadratowy x
pow	double pow(double x, double y);	x^y - x do potęgi y
sin	double sin(double x);	sinus argumentu x w radianach
atan	double atan(double x);	arcus tangens argumentu x
atan2	double atan2(double y, double x);	arcus tangens ilorazu y/x

- Większość funkcji ma po trzy wersje - dla argumentów typu: **float**, **double** i **long double**

Język C - Funkcja printf

- Ogólna składnia funkcji **printf**

```
printf("łańcuch_sterujący", arg1, arg2, ...);
```

- W najprostszej postaci **printf** wyświetla tylko tekst

```
printf("Witaj świecie");
```

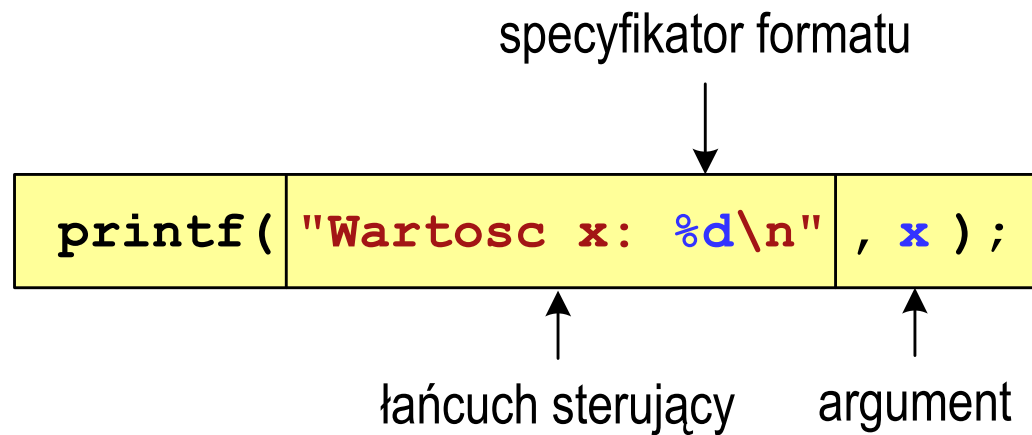
```
Witaj świecie
```

- Do wyświetlenia wartości zmiennych konieczne jest zastosowanie **specyfikatorów formatu**, określających typ oraz sposób wyświetlania argumentów

```
%[znacznik][szerokość][.precyzja][modyfikator]typ
```

Język C - Funkcja printf

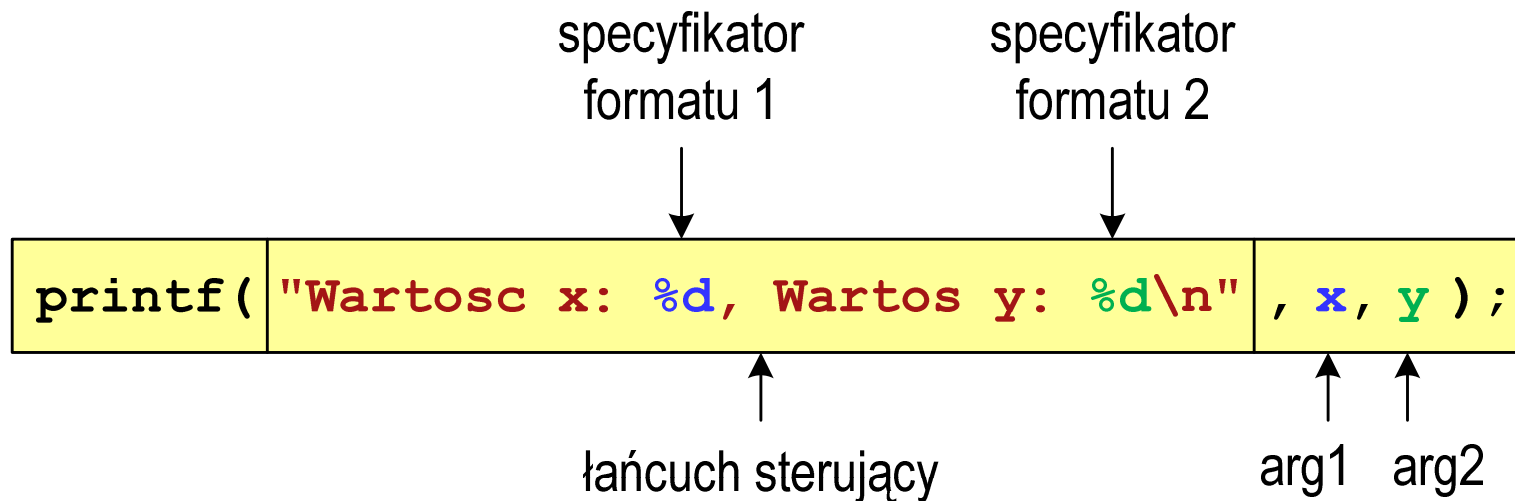
```
int x = 10;  
printf("Wartosc x: %d\n", x);
```



```
Wartosc x: 10
```

Język C - Funkcja printf

```
int x = 10, y = 20;  
printf("Wartosc x: %d, Wartosc y: %d\n", x, y);
```



```
Wartosc x: 10, Wartosc y: 20
```


Język C - Specyfikatory formatu (printf)

Typ w C	Specyfikator	Uwagi
char	%c	pojedynczy znak
	%d	kod ASCII znaku, liczba całkowita
char *	%s	łańcuch znaków, napis
int	%d %i	liczba całkowita, dziesiętna
	%o %O	liczba całkowita, ósemkowa
	%x %X	liczba całkowita, szesnastkowa
float double	%f	liczba rzeczywista
	%e %E	liczba rzeczywista, format naukowy
	%g %G	liczba rzeczywista (%f lub %e)

Język C - Funkcja printf

```
int x = 123; float y = 1.23456789f;
```

```
printf("x = [%d], y = [%f]\n", x, y);  
printf("x = [], y = []\n", x, y);  
printf("x = [%d], y = [%d]\n", x, y);
```

```
x = [123], y = [1.123457]  
x = [], y = []  
x = [123], y = [-536870912]
```

Język C - Funkcja printf

```
int x = 123; float y = 1.23456789f;
```

```
printf("x = [%6d], y = [%12f]\n", x, y);  
printf("x = [%6d], y = [%12.3f]\n", x, y);  
printf("x = [%6d], y = [%.3f]\n", x, y);
```

```
x = [ 123], y = [ 1.123457]  
x = [ 123], y = [ 1.123]  
x = [ 123], y = [1.123]
```

%[znacznik][szerokość][.precyzja][modyfikator]**typ**

Język C - Funkcja printf

```
int x = 123; float y = 1.23456789f;
```

```
printf("x = [%+6d], y = [%+12f]\n", x, y);  
printf("x = [%-6d], y = [%-12f]\n", x, y);  
printf("x = [%06d], y = [%012f]\n", x, y);
```

```
x = [ +123], y = [ +1.123457]  
x = [123   ], y = [1.123457   ]  
x = [000123], y = [00001.123457]
```

%[znacznik][szerokość][.precyzja][modyfikator]typ

Język C - Funkcja printf

```
int x = 123; float y = 1.23456789f;
```

```
printf("x = [%d], y = [%f]\n", x, y);  
printf("x = [%d], y = [%f]\n", x+321, y*25.5f);  
printf("x = [%d], y = [%f]\n", 123, 2.0f*sqrt(y));
```

```
x = [123], y = [1.123457]  
x = [444], y = [28.648149]  
x = [123], y = [2.119865]
```

Język C - Funkcja scanf

- Ogólna składnia funkcji `scanf`

```
scanf ("specyfikator", adresy_argumentów) ;
```

- Składnia `specyfikatora formatu`

```
% [szerokość] [modyfikator] typ
```

- Argumenty są adresami obszarów pamięci, dlatego muszą być poprzedzone znakiem `&`

```
int x;  
scanf ("%d", &x) ;
```

Język C - Funkcja scanf

- **Specyfikatory formatu** w większości przypadków są takie same jak w przypadku funkcji **printf**
- Największa różnica dotyczy typów **float i double**

Typ w C	Specyfikator	Uwagi
float	%f	liczba rzeczywista
	%e %E	liczba rzeczywista, format naukowy
	%g %G	liczba rzeczywista (%f lub %e)
double	%lf	liczba rzeczywista
	%le %LE	liczba rzeczywista, format naukowy
	%lg %LG	liczba rzeczywista (%f lub %e)

Język C - Funkcja scanf

```
int a, b, c;  
scanf("%d %d %d", &a, &b, &c);
```

- Wczytywane argumenty mogą być oddzielone od siebie dowolną liczbą białych (niedrukowalnych) znaków: **spacja**, **tabulacja**, **enter**

15 20 -30

15 20 -30<enter>

15 20 -30

15 20 -30<enter>

15
20
-30

15<enter>
20<enter>
-30<enter>

Przykład: częstotliwość rezonansowa

```
#include <stdio.h>
#define _USE_MATH_DEFINES
#include <math.h>

int main(void)
{
    double R, L, C, fr;

    printf("Podaj R [Om]: "); scanf("%lf", &R);
    printf("Podaj L [H]: "); scanf("%lf", &L);
    printf("Podaj C [F]: "); scanf("%lf", &C);

    fr = 1/(2*M_PI*sqrt(L*C));

    printf("-----\n");
    printf("fr [Hz]: %.3f\n", fr);

    return 0;
}
```

```
Podaj R [Om]: 100
Podaj L [H]: 0.01
Podaj C [F]: 1e-6
-----
fr [Hz]: 1591.549
```

$$f_r = \frac{1}{2\pi\sqrt{LC}}$$

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 ?



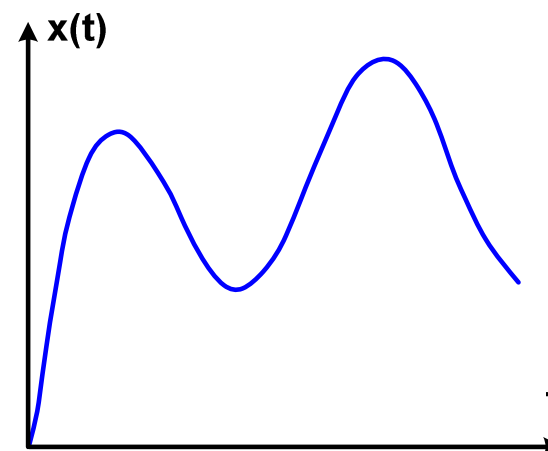
15 lipca 1410 roku

Data !!!

Informacja analogowa i cyfrowa

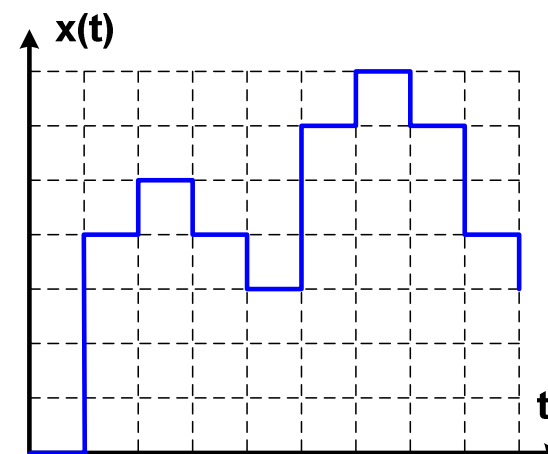
■ Sygnał analogowy

- 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

I	V	X	L	C	D	M
<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)	⊙	(Φ)
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	Ϛ	ϛ	η	Ϡ
1000	2000	3000	4000	5000	6000	7000	8000	9000
ϠͰϚβ								

cyfry w pisowni chińskiej

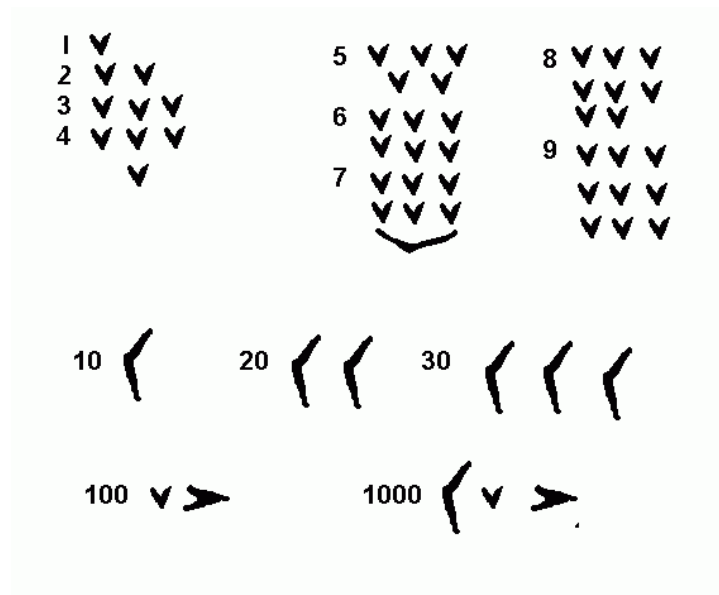
jeden	一	sześć	六
dwa	二	siedem	七
trzy	三	osiem	八
cztery	四	dziewięć	九
pięć	五	dziesięć	十
zero	另		

uczniechinskiego.com

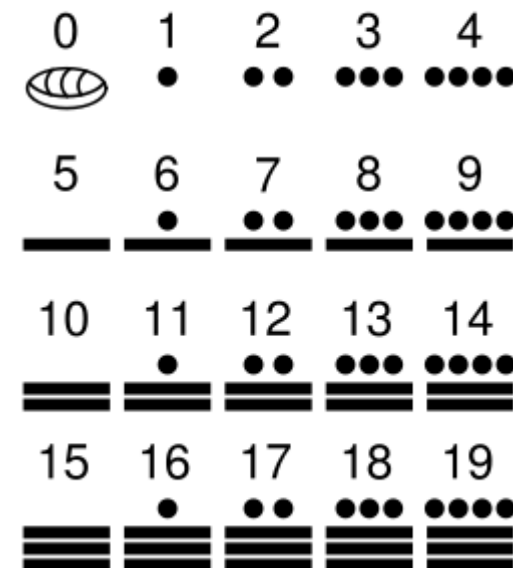
Liczby i cyfry

- Inne przykłady zapisu cyfr i liczb:

liczby w piśmie klinowym
(Babilończycy)



system prekolumbijski



Koniec wykładu nr 3

Dziękuję za uwagę!