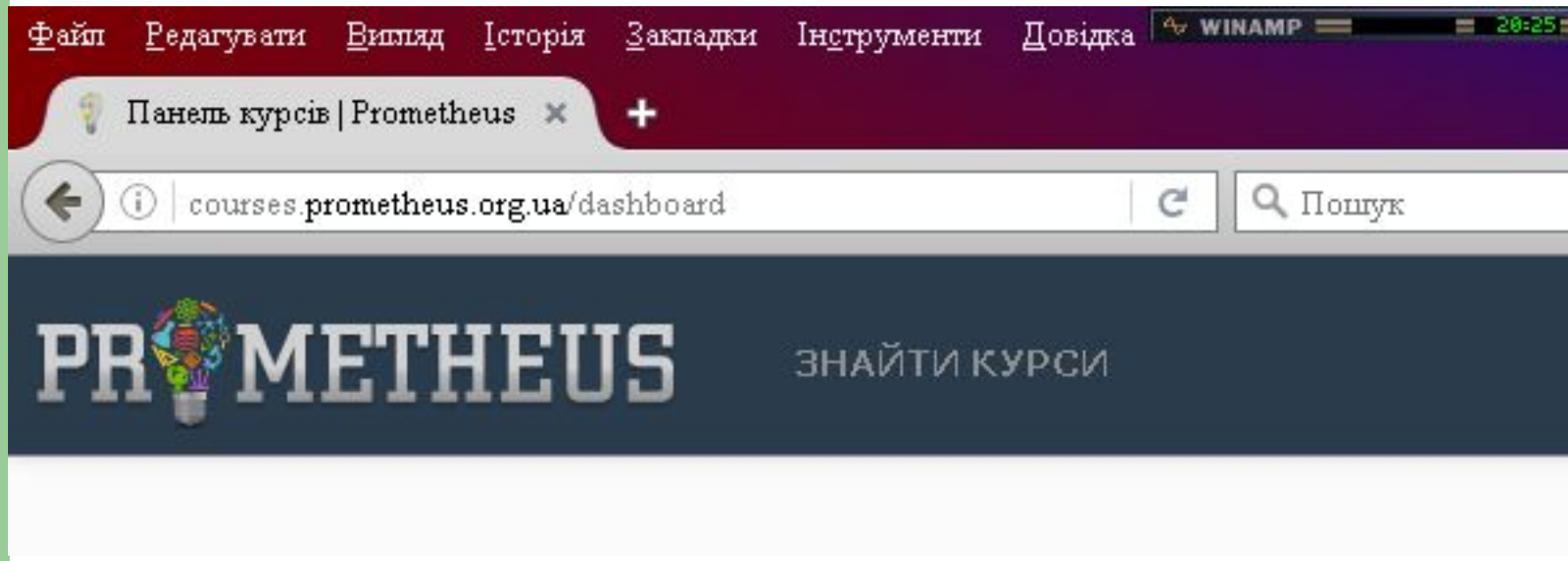
Введення в С

Квітень 2017

Віртуальна лабораторія



Hello world!

Hello world!

```
#include <stdio.h>

int main(void)
{
    printf("hello, world\n");
}
```

Hello, everyone.

```
#include <cs50.h>
#include <stdio.h>

int main(void)
{
    printf("state your name");
    string s = GetString();
    printf("hello, %s\n", s);
}
```

Основи С

Тип	Що це?	Приклад даних	Розмір (байт)
int	Цілі числа	0; 1; 2; -3; 4; 10254	4
float	Дробові числа	1.5; 5.0; 123.4567	4
double	Дробові числа	1.5; 5.0; 123.45678932	8
long long	Довгі цілі	0; 1; 2;	8
char	Символ	'A'; 'B'; 'x'; '@'	1
short	Короткі цілі	-124; 0; 1; 125	2
string	Рядки символів	"Hi"; "This is fine"	??
bool	Істина чи хибність	true; false	1 біт

Змінні

- Назви повинні складатися з літер і цифр, першим знаком має бути літера.
- Жорсткий пробіл ≪ _≫ теж вважається літерою, він часом корисний для покращення прочитності довгих назв змінних.
- Однак, не починайте назв змінних з жорсткого пробілу, оскільки функції бібліотеки часто використовують такі назви для власних потреб.

Змінні

- НЕ можна починати змінну з цифри.
- Літери верхнього регістру та літери нижнього регістру різняться, тож x та X — це дві різні назви.

Змінні

```
1  int celsium;
2  float farengate;
3  double variable;
4
5
6
7
8
9
```

Математичні операції

+	Додавання	x=x+10;
-	Віднімання/ зміна знаку числа	x= 10-y; y= - y ;
*	Множення	x=10*y;
/	Ділення (цілі діляться з відкиданням остачі)	x=y/3;
%	Остача від ділення цілих	x =21; y = 6; z = x %y; z --?
++	Оператор інкремента	x ++ або ++x
--	Декремент	x -- або –x
+=	Скорочений запис	x += 5 (x=x+5)
-=	Скорочений запис	x-= y (x=x-y)
/=	Скорочений запис	x /= y (x=x / y)
%=	Скорочений запис	x %= y (x=x % y)

Умовні оператори

```
1   if (умова)
2   {
3       //Зробити щось, якщо умова істинна
4   }
5
```

```
7   if (умова)
8   {
9       //Зробити щось, якщо умова істинна
10  }
11  else
12  {
13      //Зробити щось, якщо умова хибна
14  }
```

Приклад застосування if/else

```
1  if (time < 12)
2  {
3    printf("Доброго ранку!\n");
4  }
5  else
6  {
7    printf("Доброго дня!\n");
8  }
9
```

- Розглянути приклади у віртуальній лабораторії
- logic & positive

Логічні операції

Операція	Символ	Приклад	Коли повертає значення true (істина)
Рівність	==	a == b	Значення a і b однакові
Не дорівнює	!=	a != b	Значення a і b різні
Менше	<	a < b	Значення a менше ніж b
Менше або рівне	<=	a <= b	Значення a менше або рівне b
Більше	>	a > b	Значення a більше ніж b
Більше або рівне	>=	a >= b	Значення a більше або рівне b
Логічне «І»	&&	a && b	Коли a – true, b – true.
Логічне «АБО»	||	a || b	Коли або a – true, або b – true.

Оператор вибору case

```
switch (n)
{
  case константа1:
    // виконати, якщо n дорівнює константі1
    break;
  case константа2:
    // виконати, якщо n дорівнює константі2
    break;
   ...
  default:
    // виконати, якщо n не дорівнює жодній із вказаних констант.
    break;
}
```

Оператор вибору case

```
1   switch (n)
2   {
3     case 50:
4       printf("CS50 is Introduction to Computer Science I\n");
5       break;
6     case 51:
7       printf("CS51 is Introduction to Computer Science II\n");
8       break;
9     default:
10      printf("Sorry, I'm not familiar with that class!\n");
11      break;
12  }
```

Цикли

```
1  while (умова)
2  {
3      // виконувати поки умова істинна
4  }
5
```

```
1  do
2  {
3      // виконувати поки умова істинна
4  }
5  while (умова);
6
```

Цикли

```
lower = 0; /* нижня межа температурної шкали *
upper = 300; /* верхня межа */
step = 20; /* розмір кроку */
fahr = lower;
while (fahr <= upper) {
    celsius = (5.0/9.0) * (fahr-32.0);
    printf("%3.0f %6.1f\n", fahr, celsius);
    fahr = fahr + step;
}
```

Показати temperatere00.c

Цикл for повторює дію певну кількість разів.

```
for (int dwarves = 0; dwarves < 7; dwarves++)
{
  printf("I'm here to help you, Snow White!\n");
}
```

```
Цикли можна вкладати один в одний:
for (int row = 0; row < 3; row++)
{
  for (int column = 0; column < 4; column++)
  {
    printf("x");
  }
  printf("\n");
}
```

Показати temperature0 & temperature1

Каманди консолі

```
cd

ls

mkdir

rm

rmdir

...
```

Введення з клавіатури

```
get_char

get_double

get_float

get_int

get_long_long

get_string
```

Hello, world

```c
#include <stdio.h>
main()
{
    printf("hello, world\n");
}
```

Послідовність знаків \n означає в C символ нового рядка, який під час виводу переводить ланцюжок тексту на новий рядок.

```c
#include <stdio.h>
main()
{
    printf("hello, ");
    printf("world");
    printf("\n");
}
```

Приклади

```
printf (“%d”, 12);
printf(“Rezult is: x=%i.”, 10/12);
printf(“Rezult is: x=%f.”, 10/9);
printf(“Rezult is: x=%3.2f .”, 10/9);
```

Виведення в консолі (printf)

...printf Conversion-Type Characters

The information in this table is based on the assumption that no flag characters, width specifiers, precision specifiers, or input-size modifiers were included in the format specifier.

- NOTE: Certain conventions accompany some of these format specifiers.

Type Char	Expected Input	Format of output
Numerics		
d	Integer	Signed decimal integer
i	Integer	Signed decimal integer
o	Integer	Unsigned octal integer
u	Integer	Unsigned decimal integer
x	Integer	Unsigned hexadecimal int (with a, b, c, d, e, f)
X	Integer	Unsigned hexadecimal int (with A, B, C, D, E, F)
f	Floating point	Signed value of the form [-]dddd.dddd.

Виведення в консолі (printf)

Help

X	Integer	Unsigned hexadecimal int (with A, B, C, D, E, F)
f	Floating point	Signed value of the form [-]dddd.dddd.
e	Floating point	Signed value of the form [-]d.dddd or e[+/-]ddd
g	Floating point	Signed value in either e or f form, based on given value and precision. Trailing zeros and the decimal point are printed if necessary.
E	Floating point	Same as e; with E for exponent.
G	Floating point	Same as g; with E for exponent if e format used

Characters

c	Character	Single character
s	String pointer	Prints characters until a null-terminator is pressed or precision is reached
%	None	Prints the % character

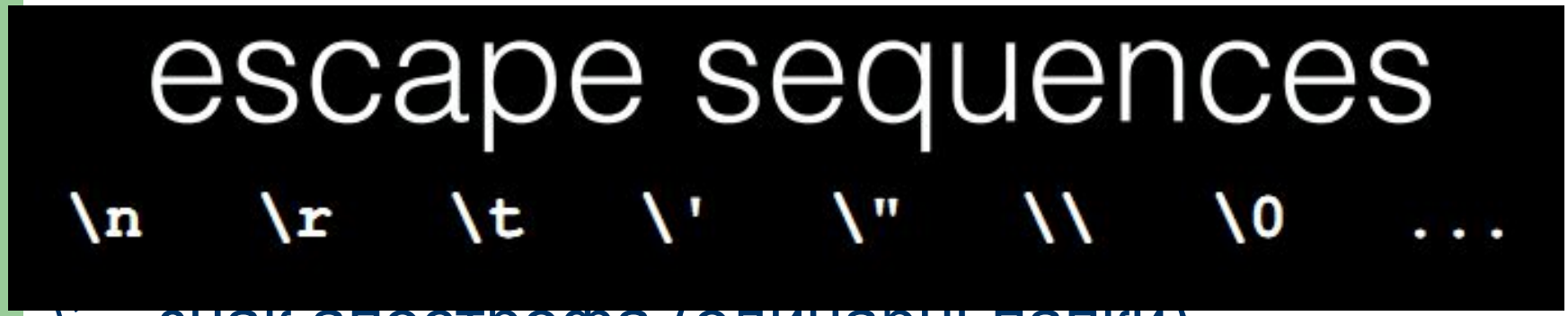
Pointers

n	Pointer to int	Stores (in the location pointed to by the input argument) a count of the chars written so far.
p	Pointer	Prints the input argument as a pointer; format

Приклади

- Програма ….. ;)

Іскейп послідовності



\' – знак апострофа (одинарні лапки)

\” – знак подвійних лапок

\\ - знак зворотнього слеша

\0 – нуль термінатор (службове значення)

- Звідки брати завдання ?

Практичне завдання

- Написати програму для введення 2 чисел і виведення більшого з них.
- Написати програму виведення додатних чисел до заданого з клавіатури.
- Написати програму виведення непарних (парних/ простих) чисел до заданого з клавіатури.