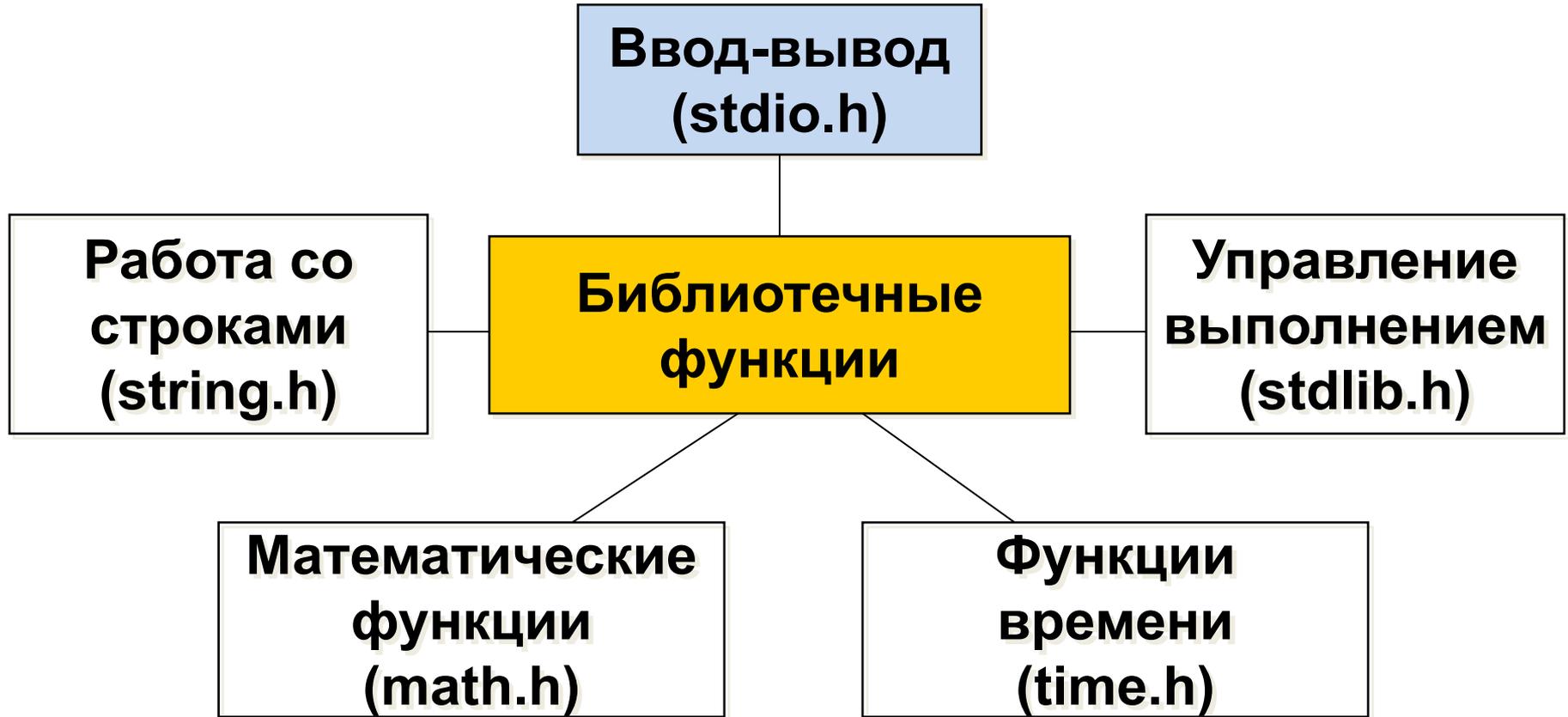


Тема 3.
Основные функции
ввода/вывода

Системная библиотека С



Системная библиотека С предоставляет программисту широкий набор функций, предназначенных для выполнения в программах типовых действий.

Основные функции ввода/вывода³ в стиле C

```
int scanf (const char* format, ...) //ввод  
int printf (const char* format, ...) //вывод
```

Функция `printf` выводит в стандартный поток вывода строку отформатированную в соответствии с правилами, указанными в строке, на которую указывает аргумент `format`.

Правила задаются набором трех типов директив:

1. Обычные символы (кроме '%' и '\'), которые выводятся без изменения;
2. Спецификаторы формата;
3. Специальные символы.

Спецификаторы формата

Каждый спецификатор начинается со знака процента %.
После % идут компоненты формата в следующем порядке:

% [флаги] [ширина] [. точность] [модификаторы] тип

	Назначение
флаги	Выравнивание, знак +
ширина	Минимальное число символов поля
точность	Максимальное число символов поля, для целых – минимальное число выводимых цифр
модификатор	Уточняет размер типа переменной
тип	Тип переменной

Флаги

Спецификатор формата может иметь несколько **флагов**, которые могут принимать значения указанные в таблице

Флаг	Назначение
–	Выравнивание результата по левой границе, если не задан, то результат выравнивается по правой границе
+	Обязательно выводятся знаки + для положительных чисел
‘ ‘	Если не указан модификатор '+', то перед положительными числами, на месте знака числа, будет выведен пробел
#	Альтернативная форма представления некоторых типов

Спецификатор **флаги** можно не указывать.

Ширина

Ширина	Назначение
n	Печатается как минимум n знаков. Если число имеет меньше символов, то дополняется пробелами
0n	Печатается как минимум n знаков. Если число имеет меньше символов, то дополняется нулями
*	Ширина указывается переменной типа int в списке аргументов

Спецификатор **ширина** можно не указывать.

Точность

Точность	Назначение
нет	Используется точность по умолчанию
.0	Децимальный знак не печатается
.n	Печатается n децимальных знаков. Для формата s задает максимальное число символов
*	Точность указывается переменной типа int в списке аргументов

Действия спецификатора **точность** зависит от типа выводимого числа.

Спецификатор **точность** можно не указывать.

Модификаторы

Префикс	Типы	Назначение
h	d i o u x X	short int
l	d i o u x X	long int
l	e E f g G	double
L	e E f g G	long double

Спецификатор **модификаторы** определяет размер выводимых данных (short, long, double, long double).

Типы преобразования

Тип	Назначение
i или d	Знаковое десятичное целое
o	Беззнаковое восьмеричное целое
u	Беззнаковое десятичное целое
x, X	Беззнаковое шестнадцатеричное целое
f, F	Знаковое число в формате с плавающей точкой
e, E	Знаковое число в экспоненциальном формате
g, G	Знаковое число в смешанном формате
c	Один символ
s	Печатается строка символов до нулевого

Специальные символы

Тип	Назначение
<code>\a</code>	Издаётся звуковой сигнал
<code>\b</code>	Удаляет последний выведенный символ
<code>\f</code>	Перевод строки. Новый символ будет напечатан на позиции, следующей за последним символом
<code>\n</code>	Перевод строки. Следующий символ будет напечатан с начала новой строки.
<code>\v</code>	Вертикальная табуляция
<code>\t</code>	Табуляция по горизонтали
<code>\\</code>	Вывод обратного слеша
<code>\'</code>	Вывод кавычек
<code>\%</code>	Вывод процента

Примеры

```
{ int d1=123, d2=42, d3=1543;  
printf ("%6d\n%6d\n%6d\n", d1, d2, d3); }
```

```
{ int d1=123, d2=42, d3=1543;  
printf ("% -6d\n% -6d\n% -6d\n", d1, d2, d3); }
```

```
{ double d1=123.4567, d2=1234567.34567;  
printf ("%lg\n%lg\n", d1, d2); }
```

```
{ double d1=123.456;  
printf ("% .21f\n", d1); }
```