

# Семинар №10

ФАКИ 2017

---

Бирюков В. А.

November 17, 2017

# Форматированный ввод/вывод

---

# Функции printf(), scanf() и sprintf(), sscanf()

Считывание и печать даты в формате: YYYY-MM-DD.

Например "2008-10-5".

Из стандартного входа в стандартный выход:

```
int year, month, day;  
printf("%d-%2d-%2d", 2008, 10, 5);  
scanf("%d-%d-%d", &year, &month, &year);
```

Из строки в строку:

```
char str1[20];  
sprintf(str1, "%d-%2d-%2d", 2008, 10, 5);  
char str2[20] = "2010-5-2";  
sscanf(str2, "%d-%d-%d", &year, &month, &day);
```

# Конвертация чисел в строку и наоборот

Конвертация чисел в строки с помощью sprintf():

```
float x = 687.74564;  
char str1[20];  
sprintf(str1, "%f", x);
```

Конвертация строк в числа с помощью sscanf():

```
float x;  
char str2[20] = "687.74564";  
sscanf(str2, "%f", &x);
```

# Работа с файлами

---

# Работа с файлами

## Открываем/закрываем файл

```
#include <stdio.h>

int main()
{
    FILE* fp = fopen("myfile.txt", "w");
    // ...
    fclose(fp);
}
```

```
#include <stdio.h>

int main()
{
    FILE* fp = fopen("myfile.txt", "r+");

    int year, month, day;
    fscanf(fp, "%d-%d-%d", &year, &month, &day);
    fprintf(fp, "Hello file\n");
    fprintf(fp, "%d-%2d-%2d\n", 1972, 5, 19);

    fclose(fp);
}
```

```
#include <stdio.h>

int main()
{
    FILE* fp = fopen("myfile.txt", "r+");
    if (fp == NULL)
    {
        printf("Error! Can't open file!\n");
        exit(1);
    }
    fprintf(fp, "Hello file\n");
    fclose(fp);
}
```



```
FILE* fopen(const char* filename, const char*  
mode);
```

r	открыть существующий файл для чтения
w	создать новый файл и открыть его для записи
a	открыть для записи в конец файла
r+	открыть для чтения/записи, с начала файла
w+	создать новый файл и открыть его для чтения/записи
a+	открыть для чтения/записи в конец файла

# Посимвольное считывание/запись

Функции `fputc()` и `fgetc()`

```
FILE * f = fopen("input.txt", "r");
int c, number_of_digits = 0;

while ((c = fgetc(f)) != EOF)
{
    if (c >= '0' && c <= '9')
        number_of_digits += 1;
}
fclose(f);
```

# Бинарные чтение/запись

Функции `fwrite()` и `fread()`

```
char str[] = "some string data";  
int arr[5] = {55, 66, 77, 88, 99};
```

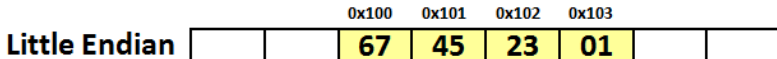
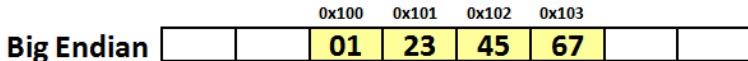
```
FILE* fp = fopen("myfile.data", "w");  
fwrite(c, strlen(c) + 1, 1, fp);  
fwrite(arr, sizeof(int), 5, fp);  
fclose(fp);
```

```
fp = fopen("myfile.data", "r");  
fread(c, strlen(c) + 1, 1, fp);  
fread(arr, sizeof(int), 5, fp);  
fclose(fp);
```

# Порядок байт little/big endian

Число 19088743 в шестнадцатеричной системе счисления представляется в виде 0x1234567.

В памяти это записывается следующим образом:



# Аргументы функции `main()`

---

## Аргументы функции main()

```
~/ $ gcc -o prog_name prog.c
```



```
argc = 4
```

```
argv = {"gcc", "-o", "prog_name", "prog.c"}
```

# Аргументы функции main()

```
int main ( int argc, char * argv[] )  
{  
    <Операторы>  
    return ( 0 );  
}
```

- argv – параметры, передаваемые в функцию main
- argc – количество этих параметров

## Сортировки за $O(n)$

---



- Сортировка подсчётом (`count sort`)
- Цифровая сортировка (`radix sort`)

# Задание

---

- Посчитать число строк, слов, символов в файле.
- Написать аналог программы wc.
- Контекст Fail not found (задачи arguments, D и cp\*)