Семинар #5: Строки. Домашнее задание.

Задача 1. Тип символа

Напишите программу, которая будет считывать символ и печатать:

- `Uppercase Letter`, если этот символ – заглавная буква
- `Lowercase Letter`, если этот символ – строчная буква.
- `Digit`, если этот символ – цифра
- `Other`, если это какой-то другой символ

Задача 2. Номер буквы

Считайте символ буквы латинского алфавита и напечатайте его номер в алфавите. Если на вход подаётся не буква, то нужно напечатать `Not a letter`.

вход	выход
P	16
b	2
B	2
#	Not a letter

Задача 3. Лесенка

Считать слово и напечатать лесенку из этого числа. Например, для слова `Hello` нужно напечатать лесенку:

вход	выход
Hello	H
	He
	Hel
	Hell
	Hello

Задача 4. Чередование

Считать 2 слова и печатать их чередуя по одному символу. То есть сначала напечатать первый символ первой строки, потом первый символ второй строки, потом второй символ первой строки, второй символ второй и т. д. Если какая-то из строк закончится, то нужно допечатать оставшуюся строку.

вход	выход
cat dog	cdaotg
cat elephant	cealtephant
elephant dog	edloegphant
aaaa bbbb	abababab
aaaa b	abaaa
a b	ab

Задача 5. Восклицание

На вход подаётся строка. Напечатать эту же строку, но ставя восклицательный знак после каждого слова.

вход	выход
Better late than never	Better! late! than! never!
cat dog elephant	cat! dog! elephant!
cat	cat!
a	a!

Задача 6. Правильная скобочная последовательность

На вход подаётся скобочная последовательность(строка, состоящая из символов '(' и ')'). Нужно выяснить является ли эта скобочная последовательность допустимой или нет.

вход	выход
(()())	Yes
((()()()))	Yes
(()))()())	No
(((())))	Yes
(((())))))	No
((((()))	No

вход	выход
()	Yes
)(	No
))	No
((	No
()()	Yes
(	No
)	No

Задача 7. Палиндром

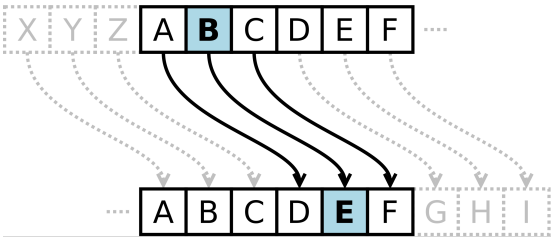
Написать функцию `is_palindrom`, которая будет принимать строку и проверять является ли эта строка палидромом.

вход	выход
cat	No
abba	Yes
aba	Yes
a	Yes
aa	Yes
ab	No

вход	выход
abcdedcba	Yes
abcdedcb	No
abcdedcbab	No
abcdedcbb	No
abcxedcba	No
abcxexcba	Yes

Задача 8. Шифр Цезаря

Шифр Цезаря — это вид шифра подстановки, в котором каждый символ заменяется символом, находящимся на некотором постоянном числе позиций левее или правее него в алфавите.



Напишите функцию `void encrypt(char* str, int k)`, которая будет зашифровывать фразу шифром Цезаря.

вход	выход
1 ABCZ	BCDA
15 ZzZzZ	OoOoO
7 The Fox Jumps Over The Dog	Aol Mve Qbtwz Vcly Aol Kvn
13 Green Terra	Terra Green

Задача 9. Самое длинное слово

Напишите функцию `int longest_word(const char* str, char* result)`, которая будет искать самое длинное слово в строке `str` и записывать его в строке `result`. Строка должна возвращать длину этого слова. Под словом тут понимается последовательность непробельных символов, ограниченная с двух сторон пробельными символами или границами строки. В случае если есть несколько слов с самой большой длиной в `result` нужно записать первое из них.

`str`	`result` после вызова `longest_word(str, result)`
cats and dogs	cats
cat dog elephant mouse	elephant
cat dog elephant mouse	elephant

Задача 10. Удаление символа

Напишите функцию `void delete_chars(char* str, char c)`, которая будет удалять все символы, равные `c` из строки `str`. Постарайтесь сделать эту функцию как можно более эффективной.

`str, c`	`str` после вызова `delete_chars(str, c)`
`cat a`	`ct`
`elephant e`	`lphant`
`aaaa a`	
`a a`	
`abababababa a`	`bbbb`

Задача 11. Сокровище

Вы находитесь на плоскости в начале координат (точке с координатами `(0, 0)`) и вам нужно найти закопанное сокровище. Путь до него задаётся последовательностью команд, состоящих из направления и расстояния, которое нужно пройти в этом направлении. Вам нужно найти координаты сокровища. Используйте функцию `strcmp`.

вход	выход

`6`
`North 10`
`East 20`
`South 50`
`West 60`
`East 10`
`North 60`
	`-30 20`

Задача 12. Безопасный `strcpy`

Известно, что функция `strcpy` небезопасна, так как может выйти за границы массива, в который она записывает. Например, в следующем примере:

```
char a[10] = "Mouse";
char b[50] = "LargeElephant";
strcpy(a, b); // UB, выйдет за пределы массива a
```

Ваша задача заключается в том, чтобы написать функцию `safe_strcpy`, которая будет более безопасной, чем функция `strcpy` и не будет выходить за границы массива. Эта функция должна иметь 3 параметра:

- строка в которую мы будем записывать (тип `char*`)
- размер массива в который мы будем записывать (тип `size_t`)
- строка из которой мы будем считывать (тип `const char*`).

В случае если строка из которой мы считываем не будет помещаться в массив, нужно скопировать только часть строки, которая может поместиться в него. И обязательно поставить нулевой символ в конце строки.

```
char a[10] = "Mouse";
char b[50] = "LargeElephant";
safe_strcpy(a, 10, b); // OK, строка a будет равна "LargeElep\0"
```

Задача 13. Повторитель

Напишите программу `repeater`, которая будет принимать через аргументы командной строки некоторое слово и некоторое число. Эта программа должна печатать это слово столько раз, чему равно переданное число. Например, если мы вызовем эту программу так:

```
./repeater Hello 5
```

то программа должна напечатать:

```
Hello Hello Hello Hello Hello
```

Задача 14. Сортировка аргументов

Напишите программу `sort`, которая будет принимать через аргументы командной строки произвольное число строк и сортировать эти строки лексикографически и печатать их.
Например, если мы вызовем эту программу так:

```
./sort cat elephant mouse axolotl lion
```

то программа должна напечатать:

```
axolotl cat elephant lion mouse
```

Задача 15. Шифрование файла

Напишите программу `encrypt`, которая будет принимать через аргументы командной строки названия входного и выходного файлов, а также число-ключ шифра Цезаря. Программа должна считывать входной файл, зашифровывать его шифром Цезаря и записывать результат в выходной файл.
Например, если мы вызовем эту программу так:

```
./encrypt three_little_pigs.txt result.txt 7
```

то программа должна считывать файл `three_little_pigs.txt`, зашифровывать содержимое шифром Цезаря с ключом 7 и записывать результат в файл `result.txt`.

Задача 16. Замена

Напишите программу `replacer`, которая будет принимать через аргументы командной строки названия входного и выходного файлов, а также две строки. Программа должна считывать входной файл, заменять все вхождения первой строки на вторую строку и записывать результат в выходной файл.
Например, если мы вызовем эту программу так:

```
./replacer three_little_pigs.txt result.txt pig elephant
```

то программа должна считывать файл `three_little_pigs.txt`, заменять все подстроки `"pig"` на `"elephant"` и записывать результат в файл `result.txt`. Постарайтесь написать эффективный алгоритм замены. Рассмотите случаи когда подстрока заменяется на более длинную строку и когда подстрока заменяется на более короткую строку.

Необязательные задачи

Задача H1: Переворот слов в файле

Счмтайте файл `input.txt` и переверните все слова из этого файла и запишите результат в файл `output.txt`.

файл `input.txt`	файл `output.txt`
`Cat and Dog`	`taC dna goD`

Протестируйте программу на файле `invisible_man.txt`.

Задача H2: to upper

Напишите функцию `void string_to_upper(char* p)`, которая будет принимать на вход указатель на первый символ строки и переводить эту строку в верхний регистр.

Задача H3: trim

Напишите функцию `void trim_after_first_space(char* p)`, которая будет принимать на вход указатель на первый символ строки и укорачивать строку до первого пробела. Протестируйте функции, вызвав её их из функции `main` с помощью следующего кода:

```c
#include <stdio.h>
// Тут вам нужно написать функции string_to_upper и trim_after_first_space
int main()
{
    char a[] = "Sapere Aude";
    string_to_upper(a);
    printf("%s\n", a); // Должно напечатать SAPERE AUDE
    trim_after_first_space(a);
    printf("%s\n", a); // Должно напечатать SAPERE
}
```