



Submit a solution for up05-1-mz05-1 (дореш)

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/10/09 10:00:00
Deadline:	2022/10/23 10:00:00
Date penalty formula:	2022/12/31 0-50/7d
Current penalty:	50

Problem up05-1: mz05-1 (дореш)

В аргументах командной строки передаются пути к файлам.

На стандартный поток вывода напечатайте суммарный размер файлов, удовлетворяющих одновременно следующим условиям:

- Файл существует и его размер равен целому числу кибибайт (2^{10}).
- Файл является регулярным, при этом не является символической ссылкой.
- Файл имеет единственное имя в файловой системе.

Submit a solution

Language: gcc-32 - GNU C (32 bit) 12.1.1

File Файл не выбран

Send!

mz01-1

mz01-2

mz01-3

mz01-4

mz01-5

mz01-6

up01-1

up01-2

up01-3

up01-4

up01-5

up01-6

mz02-1

mz02-2

mz02-3

mz02-4

mz02-5

mz02-6

up02-1

up02-2

up02-3

up02-4

up02-5

up02-6

up02-7

mz03-1

mz03-2

mz03-3

mz03-4

mz03-5

up03-1

up03-2

up03-3

up03-4

up03-5

mz04-1

mz04-2

mz04-3

mz04-4

mz04-5

mz04-6

up04-1

up04-2

up04-3

up04-4

up04-5

mz05-1

mz05-2

mz05-3

mz05-4

mz05-5

up05-1

up05-2

up05-3

up05-4

up05-5

mz06-1

mz06-2

mz06-3

mz06-4

mz06-5

up06-1

up06-2

up06-3

up06-4



Submit a solution for up05-2-mz05-2 (дореш)

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/10/09 10:00:00
Deadline:	2022/10/23 10:00:00
Date penalty formula:	2022/12/31 0-50/7d
Current penalty:	50

Problem up05-2: mz05-2 (дореш)

В аргументах командной строки передаются 9-битные восьмеричные числа, соответствующие правам доступа к файлам в UNIX. Каждый аргумент командной строки содержит только одно число. Незначащие нули у восьмеричного числа в могут отсутствовать, а могут и присутствовать. На стандартный поток вывода напечатайте права доступа в gwx-записи. Каждую строку прав доступа выводите на отдельной строке текста.

Для получения строки-результата используйте специально посвященную этому константную строку-шаблон gwxgwxgwx.

Examples

Input

644 755

Output

rw-r--r--
rwxr-xr-x

Submit a solution

Language: gcc-32 - GNU C (32 bit) 12.1.1

File Файл не выбран

Send!

mz01-1

mz01-2

mz01-3

mz01-4

mz01-5

mz01-6

up01-1

up01-2

up01-3

up01-4

up01-5

up01-6

mz02-1

mz02-2

mz02-3

mz02-4

mz02-5

mz02-6

up02-1

up02-2

up02-3

up02-4

up02-5

up02-6

up02-7

mz03-1

mz03-2

mz03-3

mz03-4

mz03-5

up03-1

up03-2

up03-3

up03-4

up03-5

mz04-1

mz04-2

mz04-3

mz04-4

mz04-5

mz04-6

up04-1

up04-2

up04-3

up04-4

up04-5

mz05-1

mz05-2

mz05-3

mz05-4

mz05-5

up05-1

up05-2

up05-3

up05-4

up05-5

mz06-1

mz06-2

mz06-3

mz06-4

mz06-5

up06-1

up06-2

up06-3

up06-4



Submit a solution for up05-3-mz05-3 (дореш)

Full score:	100
Run penalty:	10
Time limit:	2 s
Real time limit:	5 s
Memory limit:	64M
Memory RSS limit:	512M
Stack limit:	8M
Open date:	2022/10/09 10:00:00
Deadline:	2022/10/23 10:00:00
Date penalty formula:	2022/12/31 0-50/7d
Current penalty:	50

Problem up05-3: mz05-3 (дореш)

Напишите функцию

```
int
parse_rwx_permissions(const char *str);
```

На вход функции подается строка — права доступа к файлу в формате `gwx` (9 знаков). Указатель может принимать значение `NULL`.

Функция должна вернуть соответствующее значение прав доступа, либо `-1`, если строка не является корректной записью прав доступа. Корректная запись состоит только из символов `'r'`, `'w'`, `'x'`, расположенных в правильном порядке.

Указание: используйте строку-образец `"gwxgwxgwx"` для разбора входной строки. В функции не должны использоваться константы `3`, `8`, `9`, `10` (даже в виде `enum` или `#define`). Используйте битовые операции.

Поскольку строка-образец является константной, ее длину можно вычислить за константное время на этапе компиляции. В функции может использоваться не более одного вызова `strlen`.

Submit a solution

Language: gcc-vg - GNU C (valgrind) 12.1.1

File Файл не выбран

Send!

Previous submissions of this problem

Run ID	Time	Size	Problem	Language	Result	Tests passed	Score	View source	View report
1908	2022/10/12 23:24:50	519	up05-3	gcc-vg	Pending review	13	80=100-2*10	View	View
1906	2022/10/12 23:24:13	518	up05-3	gcc-vg	Compilation error	N/A	N/A	View	View
1902	2022/10/12 23:03:40	464	up05-3	gcc-vg	Partial solution	2	0=0-1*10	View	View
1901	2022/10/12 22:51:36	462	up05-3	gcc-vg	Partial solution	2	0	View	View

mz01-1

mz01-2

mz01-3

mz01-4

mz01-5

mz01-6

up01-1

up01-2

up01-3

up01-4

up01-5

up01-6

mz02-1

mz02-2

mz02-3

mz02-4

mz02-5

mz02-6

up02-1

up02-2

up02-3

up02-4

up02-5

up02-6

up02-7

mz03-1

mz03-2

mz03-3

mz03-4

mz03-5

up03-1

up03-2

up03-3

up03-4

up03-5

mz04-1

mz04-2

mz04-3

mz04-4

mz04-5

mz04-6

up04-1

up04-2

up04-3

up04-4

up04-5

mz05-1

mz05-2

mz05-3

mz05-4

mz05-5

up05-1

up05-2

up05-3

up05-4

up05-5

mz06-1

mz06-2

mz06-3

mz06-4

mz06-5

up06-1

up06-2

up06-3

up06-4



Problem mz05-4-mz05-4

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/10/03 10:30:00
Deadline:	2022/10/03 12:05:00

Problem mz05-4: mz05-4

В аргументах командной строки передаются пути к файлам.

Если в списке путей к файлам несколько раз встречается один и тот же файл, то оставьте из этих путей один лексикографически максимальный (в смысле сравнения строк) путь. Все оставшиеся (то есть уникальные) пути выведите на стандартный поток вывода в порядке лексикографического возрастания по одному пути на строку.

Если информация о файле по указанному пути недоступна, такой путь игнорируется и вывод не попадает.

Два пути являются путями к одному и тому же файлу, если либо один из них является символической ссылкой на другой, либо оба из них являются символическими ссылками на третий файл, либо файлы являются жесткими ссылками друг друга, либо пути равны.

Например, если файл В является символической ссылкой на файл А, а файл С не связан с ними, то при запуске программы с аргументами:

```
./solution С А В В
```

на стандартный поток вывода должно быть выведено

```
В
С
```

Задача будет тестироваться в офф-лайн режиме.

mz01-1

mz01-2

mz01-3

mz01-4

mz01-5

mz01-6

up01-1

up01-2

up01-3

up01-4

up01-5

up01-6

mz02-1

mz02-2

mz02-3

mz02-4

mz02-5

mz02-6

up02-1

up02-2

up02-3

up02-4

up02-5

up02-6

up02-7

mz03-1

mz03-2

mz03-3

mz03-4

mz03-5

up03-1

up03-2

up03-3

up03-4

up03-5

mz04-1

mz04-2

mz04-3

mz04-4

mz04-5

mz04-6

up04-1

up04-2

up04-3

up04-4

up04-5

mz05-1

mz05-2

mz05-3

mz05-4

mz05-5

up05-1

up05-2

up05-3

up05-4

up05-5

mz06-1

mz06-2

mz06-3

mz06-4

mz06-5

up06-1

up06-2

up06-3

up06-4

up06-5

mz07-1

mz07-2

mz07-3

mz07-4

mz07-5

up07-1

up07-2

up07-3

up07-4

up07-5

mz08-1

mz08-2

mz08-3

mz08-4

mz08-5



Problem mz05-5-mz05-5

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/10/03 10:30:00
Deadline:	2022/10/03 12:05:00

Problem mz05-5: mz05-5

Напишите функцию:

```
int copy_file(const char *srcpath, const char *dstpath);
```

srcpath должен быть путем к существующему файлу, доступному на чтение. srcpath не должен быть каталогом.

Если dstpath существует и является каталогом, то файл srcpath должен быть скопирован в каталог dstpath, при этом имя файла в каталоге dstpath должно совпадать с последней компонентой пути srcpath.

Если dstpath не существует или не является каталогом, файл dstpath перезаписывается.

Должна быть предусмотрена защита в случае, когда входной файл совпадает с выходным файлом. В этом случае должно возвращаться 0 и не выполняться никаких действий.

При успешном выполнении функция должна возвращать 0. При ошибке функция должна возвращать -1. Все ресурсы должны быть освобождены.

Для копирования файлов должны должны использоваться системные вызовы read, write. Для временного хранения данных должен использоваться внутренний буфер размером 4 кб.

Права доступа к файлу-копии должны совпадать с правами доступа к исходному файлу.

Символические ссылки специальным образом не обрабатывать.

mz01-1
mz01-2
mz01-3
mz01-4
mz01-5
mz01-6
up01-1
up01-2
up01-3
up01-4
up01-5
up01-6
mz02-1
mz02-2
mz02-3
mz02-4
mz02-5
mz02-6
up02-1
up02-2
up02-3
up02-4
up02-5
up02-6
up02-7
mz03-1
mz03-2
mz03-3
mz03-4
mz03-5
up03-1
up03-2
up03-3
up03-4
up03-5
mz04-1
mz04-2
mz04-3
mz04-4
mz04-5
mz04-6
up04-1
up04-2
up04-3
up04-4
up04-5
mz05-1
mz05-2
mz05-3
mz05-4
mz05-5
up05-1
up05-2
up05-3
up05-4
up05-5
mz06-1
mz06-2
mz06-3
mz06-4
mz06-5
up06-1
up06-2
up06-3
up06-4
up06-5
mz07-1
mz07-2
mz07-3
mz07-4
mz07-5
up07-1
up07-2
up07-3
up07-4
up07-5
mz08-1
mz08-2
mz08-3
mz08-4
mz08-5