



11:56:47 / RUNNING

Submit a solution for mz12-1-mz12-1

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/14 10:30:00
Deadline:	2022/11/14 12:05:00

Problem mz12-1: mz12-1

В аргументах командной строки передаются параметры `CMD FILE1 FILE2 FILE3`. Реализуйте запуск процесса `CMD` с перенаправлением стандартных потоков как в команде:

```
CMD < FILE1 >> FILE2 2> FILE3
```

Файлы `FILE2`, `FILE3` должны создаваться с помощью системного вызова `open` с правами чтение-запись для владельца и группы. Если файл уже существует, его права доступа не изменяются. Если в созданном в рамках вашей программы процессе-сыне какой-либо системный вызов завершился с ошибкой, этот процесс должен завершиться с кодом 42.

После завершения созданного процесса родитель должен напечатать на стандартный поток вывода целое число - статус завершения процесса, заполненное в результате системного вызова `wait` без каких-либо модификаций. Процесс-родитель должен завершаться самым последним из всех процессов. Процесс-родитель всегда завершается с кодом 0.

Для запуска файла на выполнение используйте `exec1p`. Не используйте `umask` для изменения соответствующего параметра процесса.

Например, при запуске вашей программы:

```
./prog wc in.txt out.txt err.txt
```

В отдельном процессе должна быть запущена программа `wc`, ее стандартный ввод должен быть перенаправлен на чтение из файла `in.txt`, стандартный вывод - на добавление в файл `out.txt`, а стандартный поток ошибок - на перезапись в файл `err.txt`.

Submit a solution

Language: gcc-32 - GNU C (32 bit) 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
4091	2022/11/14 11:41:10	859	mz12-1	gcc-32	Pending review	12	80=100-2*10	View	View
4087	2022/11/14 11:39:55	860	mz12-1	gcc-32	Partial solution	4	0=0-1*10	View	View
4074	2022/11/14 11:31:26	871	mz12-1	gcc-32	Partial solution	0	0	View	View
4066	2022/11/14 11:29:06	745	mz12-1	gcc-32	Coding style violation	N/A	N/A	View	View

View
[all](#) / [necessary](#)

up02-4

up02-5

up02-6

up02-7

mz09-1

mz09-2

mz09-3

mz09-4

mz09-5

mz09-6

ht10-1

ht10-2

ht10-3

ht10-4

mz12-1

mz12-2

mz12-3

mz12-4

mz12-5



11:57:00 / RUNNING

Submit a solution for mz12-2-mz12-2

Full score:	100
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/14 10:30:00
Deadline:	2022/11/14 12:05:00

Problem mz12-2: mz12-2

Программе в аргументах командной строки передаются три команды для выполнения. Программа должна запустить на выполнение процессы, чтобы они выполнялись в конфигурации, соответствующей команде shell:

```
{ cmd1 || cmd2 } && cmd3
```

Родительский процесс должен дожидаться завершения всех созданных им процессов и сам завершить выполнение с кодом возврата 0 или 1 в зависимости от успешности выполнения процессов.

Команды cmd1, cmd2, cmd3 можно запускать на выполнение с помощью exelpr.

Запись a || b означает то же самое, что !a?b:true. Запись a && b означает то же самое, что a?b:false. Скобки { и } используются для группировки.

Для шелла процесс считается завершившимся успешно, если процесс завершился с помощью системного вызова exit (макрос WIFEXITED возвращает истинное значение), и код завершения процесса (макрос WEXITSTATUS) равен 0. В других случаях процесс считается завершившимся неуспешно.

Соответственно, ваша программа должна завершиться с кодом 0 в случае успешного выполнения заданной группы процессов и с кодом 1 в случае неуспешного их выполнения.

Организуйте вашу программу таким образом, чтобы избежать дублирование кода. В частности, запуск процесса и проверка статуса его завершения может быть вынесена в функцию.

Примечания по тестированию вашей программы:. Тестирование завершается с вердиктом 'Synchronization error', если процесс-отец (то есть ваша программа, запускаемая на тестирование) заканчивает работу раньше какого-либо из своих потомков.

Submit a solution

Language:gcc-32 - GNU C (32 bit) 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
4118	2022/11/14 11:55:52	433	mz12-2	gcc-32	Pending review	10	100	View	View
4116	2022/11/14 11:54:46	405	mz12-2	gcc-32	Partial solution	0	0	View	View
4115	2022/11/14 11:54:20	391	mz12-2	gcc-32	Coding violation style	N/A	N/A	View	View

View
[all](#) / [necessary](#)

up02-4

up02-5

up02-6

up02-7

mz09-1

mz09-2

mz09-3

mz09-4

mz09-5

mz09-6

ht10-1

ht10-2

ht10-3

ht10-4

mz12-1

mz12-2

mz12-3

mz12-4

mz12-5



11:57:12 / RUNNING

Submit a solution for mz12-3-mz12-3

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	1 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/14 10:30:00
Deadline:	2022/11/14 12:05:00

Problem mz12-3: mz12-3

Программе в аргументах командной строки передаются программы для запуска. Программы могут запускаться либо параллельно, либо последовательно. Первая буква 'p' в аргументе командной строки означает, что эта программа должна работать параллельно с другими программами, помеченными 'p'. Первая буква 's' обозначает, что эта программа работает последовательно, то есть запускается на выполнение после того, как все программы в предыдущих аргументах командной строки завершены. При запуске программы первая буква в аргументе командной строки отбрасывается.

Например, если заданы аргументы "pprog1", "pprog2", "sprog3", "pprog4", "pprog5", то значала должны параллельно выполняться программы prog1 и prog2, затем после их завершения должна выполняться программа prog3, затем после ее завершения должны параллельно выполняться программы prog4 и prog5. Кавычки в аргументах командной строки отсутствуют.

Главная программа должна дожидаться завершения выполнения всех процессов и вывести на стандартный поток вывода число программ, отработавших успешно. Процесс считается отработавшим успешно, если он завершился с кодом завершения 0. Программы запускать на выполнение с помощью exesclp. Предполагать, что все системные вызовы завершаются успешно, нельзя.

Submit a solution

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

File Файл не выбран

Send!

View
[all](#) / [necessary](#)

up02-4

up02-5

up02-6

up02-7

mz09-1

mz09-2

mz09-3

mz09-4

mz09-5

mz09-6

ht10-1

ht10-2

ht10-3

ht10-4

mz12-1

mz12-2

mz12-3

mz12-4

mz12-5



11:57:32 / RUNNING

Submit a solution for mz12-4-mz12-4

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/14 10:30:00
Deadline:	2022/11/14 12:05:00

Problem mz12-4: mz12-4

В аргументах командной строки передаются целые положительные числа произвольного размера. Число передаваемых аргументов больше или равно 1. На стандартный поток вывода напечатайте произведение этих чисел.

Для вычисления результата используйте скриптовый язык программирования, поддерживающий арифметику произвольной точности, например, python, python3.

На языке python3 умножение двух чисел выглядит следующим образом:

```
print(1*2)
```

Если записать эту программу в файл `apb.py` и запустить интерпретатор командой

```
python3 apb.py
```

То на стандартном потоке вывода будет получен результат:

```
2
```

Ваша программа должна создать скрипт, который не будет требовать никаких дополнительных входных данных и при запуске выведет на стандартный поток вывода произведение чисел. При запуске команды на выполнение из вашей программы не должно указываться явное имя интерпретатора, поэтому в скрипте используйте запись `#!`.

Не создавайте новый процесс. Ваша программа может запускаться в произвольном рабочем каталоге. Программа должна поддерживать возможность одновременной работы своих копий. Программа не должна допускать возможность изменения скрипта другими пользователями. Скрипт должен удаляться после выполнения.

Не используйте функции `tmpnam`, `tmpnam_r`, `mktemp`, `mkstemp`, `tmpfile` и аналогичные. В качестве пути к каталогу временных файлов используйте значение переменной окружения `XDGRUNTIME_DIR`, если эта переменная не установлена, используйте значение переменной окружения `TMPDIR`, если и эта переменная не установлена, используйте путь `/tmp`.

Submit a solution

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

File Файл не выбран

Send!

View
[all](#) / [necessary](#)

up02-4

up02-5

up02-6

up02-7

mz09-1

mz09-2

mz09-3

mz09-4

mz09-5

mz09-6

ht10-1

ht10-2

ht10-3

ht10-4

mz12-1

mz12-2

mz12-3

mz12-4

mz12-5



11:57:47 / RUNNING

Submit a solution for mz12-5-mz12-5

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/14 10:30:00
Deadline:	2022/11/14 12:05:00

Problem mz12-5: mz12-5

В аргументе командной строки задается выражение над 32-битной целой знаковой переменной x. Выражение удовлетворяет правилам языка Си и должно вырабатывать одно из трех значений: reject, summon и disqualify.

Например, выражение:

```
x > 5?summon:disqualify
```

вырабатывает результат summon, если x > 5, и disqualify в противном случае.

На стандартный поток ввода подается последовательность 32-битных целых знаковых значений. Для каждого значения вычислите заданное выражение и выведите результат (reject, summon или disqualify) на стандартный поток вывода. После каждого результата выведите символ перехода на новую строку.

Ниже приведен пример для выражения x > 5?summon:disqualify.

Указание. Ваша программа может использовать компилятор gcc, установленный в системе.

Не используйте библиотечные функции system, popen.

Требования к решению:

- Должна поддерживаться одновременная работа нескольких копий программы.
- Для выполнения вычислений не должен создаваться отдельный процесс.
- Посторонний пользователь не должен иметь возможность ни посмотреть сгенерированные файлы, ни изменить их.
- Программа может не иметь права на запись в текущий каталог. Создавайте файлы во временном каталоге, выбираемом по правилам, описанным в предыдущей задаче.
- Временные файлы должны удаляться.
- Для генерации .c файла должен использоваться высокоуровневый ввод-вывод. Успешность записи должна проверяться.
- При возникновении ошибки должна быть содержательная диагностика.

Examples

Input

```
2
5
6
```

Output

```
disqualify
disqualify
summon
```

Submit a solution

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

File Файл не выбран

Send!

View
[all](#) / [necessary](#)

up02-4

up02-5

up02-6

up02-7

mz09-1

mz09-2

mz09-3

mz09-4

mz09-5

mz09-6

ht10-1

ht10-2

ht10-3

ht10-4

mz12-1

mz12-2

mz12-3

mz12-4

mz12-5