



12:01:35 / RUNNING

Submit a solution for mz11-1-mz11-1

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/07 10:30:00
Deadline:	2022/11/07 12:05:00

Problem mz11-1: mz11-1

Дана функция proc:

```
int proc(void)
{
    int pid = fork();
    if (!pid) {
        write(1, "1\n", 2);
    }
    return pid;
}
```

Запишите **одно выражение** на языке C, в результате выполнения которого на стандартный поток вывода строка 1 была бы выведена 7 раз. В качестве программы сдайте текст, содержащий только это выражение. В выражении можно использовать только операции языка Си и вызов функции proc не более 4 раз.

Например, для выражения

```
proc()
```

на стандартный поток вывода будет напечатано

```
1
```

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
3706	2022/11/07 10:42:52	25	mz11-1	gcc-32	Pending review	1	100	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

mz11-1

mz11-2

mz11-3

mz11-4

mz11-5



12:01:21 / RUNNING

Submit a solution for mz11-2-mz11-2

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/07 10:30:00
Deadline:	2022/11/07 12:05:00

Problem mz11-2: mz11-2

Процесс-отец (то есть процесс, который запускается из командной строки) создает процесс-сына, который в свою очередь создает своего сына (внука главного процесса).

Процесс-отец должен вывести на стандартный поток вывода число 1, сын - число 2, а внук - число 3.

Вывод должен быть таким, чтобы всегда на стандартном потоке вывода оказывалась последовательность 3 2 1, записанная в одну строку строго с одним пробелом между числами. Пробел в начале и конце строки не допускается. В конце вывода должен находиться символ \n. Для вывода использовать средства высокого уровня.

Процесс-родитель должен завершаться самым последним из всех процессов.

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
3736	2022/11/07 10:56:04	470	mz11-2	gcc-32	Pending review	1	80=100-2*10	View	View
3733	2022/11/07 10:55:15	470	mz11-2	gcc-32	Partial solution	0	0=0-1*10	View	View
3730	2022/11/07 10:53:36	536	mz11-2	gcc-32	Partial solution	0	0	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

mz11-1

mz11-2

mz11-3

mz11-4

mz11-5



12:01:07 / RUNNING

Submit a solution for mz11-3-mz11-3

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/07 10:30:00
Deadline:	2022/11/07 12:05:00

Problem mz11-3: mz11-3

На стандартном потоке ввода вводятся три строки текста, каждая из которых содержит целое 32-битное знаковое число. Длина каждой строки — ровно 8 символов с учетом завершающего символа `\n`.

Процесс-отец создает трех сыновей, которые работают параллельно друг другу. Сыновья получают порядковые номера 1, 2, 3. Каждый процесс обрабатывает одну из строк входных данных. Каждый процесс выводит на стандартный поток вывода свой порядковый номер и квадрат числа, считанного из входной строки.

Обработка заключается в чтении значения из файла, вычислении значения и выводе результата. Стандартный поток ввода не является файлом произвольного доступа (является каналом). Результат возведения в квадрат представим 32-битным знаковым типом.

Каждый результат должен быть записан на отдельной строке. Порядок вывода строк значения не имеет.

Для вывода использовать средства высокого уровня.

Процесс-родитель должен завершаться самым последним из всех процессов.

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

mz11-1

mz11-2

mz11-3

mz11-4

mz11-5



12:00:55 / RUNNING

Submit a solution for mz11-4-mz11-4

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/07 10:30:00
Deadline:	2022/11/07 12:05:00

Problem mz11-4: mz11-4

На стандартном потоке ввода задается целое число N ($N > 0$). Требуется создать $N - 1$ процессов так, чтобы первый процесс (процесс-родитель) вывел на стандартный поток вывода число 1, второй процесс (его сын) вывел на стандартный поток вывода число 2, третий процесс (сын сына) вывел 3 и так далее.

Вывод должен быть таким, чтобы всегда на стандартном потоке вывода оказывалась последовательность 1 2 3 ... N, записанная в одну строку строго с одним пробелом между числами. В конце вывода должен находиться символ $\backslash n$. Дополнительные пробелы в начале или конце строки не допускаются. Для вывода использовать средства высокого уровня.

Процесс-родитель должен завершаться самым последним из всех процессов.

Не используйте рекурсию!

Submit a solution

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

File

Выберите файл

Файл не выбран

Send!

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
mz11-1
mz11-2
mz11-3
mz11-4
mz11-5



11:38:13 / RUNNING

Submit a solution for mz11-5-mz11-5

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/07 10:30:00
Deadline:	2022/11/07 12:05:00

Problem mz11-5: mz11-5

На стандартном потоке ввода задается последовательность целых чисел типа `int`, завершающаяся признаком конца файла. Для каждого введенного числа должен создаваться отдельный процесс, который выводит его же на стандартный поток вывода, но так, что в итоге последовательность чисел на стандартном потоке вывода окажется идущей в обратном порядке относительно чисел на стандартном потоке ввода. Дополнительный процесс может фиксировать признак конца ввода на стандартном потоке ввода.

Каждый процесс отвечает за чтение, хранение и вывод только одного числа из всей последовательности.

Например, если на стандартном потоке ввода задана последовательность `1 2 3`, то в дополнение к родителю должно быть создано три процесса и на стандартный поток вывода должно быть выведено:

```
3
2
1
```

Если невозможно создать столько процессов, сколько чисел во входной последовательности, то вместо всей входной последовательности на стандартный поток вывода должно быть выведено число `-1`.

Стандартный поток ввода не является файлом произвольного доступа (является каналом).

Процесс-родитель должен завершаться самым последним из всех процессов. Этот процесс должен завершаться с кодом завершения `0` в любом случае.

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
3785	2022/11/07 11:38:09	615	mz11-5	gcc-32	Compilation error	N/A	N/A	View	View
3770	2022/11/07 11:30:00	421	mz11-5	gcc-32	Partial solution	2	0	View	View
3769	2022/11/07 11:29:06	409	mz11-5	gcc-32	Compilation error	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

mz11-1

mz11-2

mz11-3

mz11-4

mz11-5