



12:21:32 / RUNNING

Submit a solution for up13-1-mz13-1 (дореш)

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	1 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022-11-24 00:00:00
Deadline:	2022-12-15 00:00:00
Date penalty formula:	2022/12/31 0-50/7d
Current penalty:	50
Next soft deadline:	2022-12-08 00:00:00

Problem up13-1: mz13-1 (дореш)

Напишите программу. Родитель создает сына, тот — внука, а тот — правнука. Правнук передает в канал текущее время, полученное с помощью системного вызова `time`, в бинарном виде (тип `time_t`).

Отец, сын и внук считывают время из канала. Процесс-отец выводит на экран строку "Y:????", где ???? заменяются на текущий год, сын — "M:??", где ?? заменяются на текущий месяц в году (от 1 до 12), число всегда выводится с двумя знаками, внук — "D:??", где ?? заменяются на номер дня в месяце, всегда выводятся с двумя знаками.

Внук должен вывести число первым, сын — вторым, а отец — третьим. Записывать в канал разрешается только правнуку. Родитель должен дождаться завершения всех процессов и завершиться сам с кодом 0.

Пример вывода:

```
D:11
M:09
Y:2001
```

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

up13-1

up14-4

mz15-1...

mz15-2...

mz15-3

mz15-4...

mz15-5...



12:22:15 / RUNNING

Submit a solution for up13-2-mz13-2 (дореш)

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	1 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022-11-24 00:00:00
Deadline:	2022-12-15 00:00:00
Date penalty formula:	2022/12/31 0-50/7d
Current penalty:	50
Next soft deadline:	2022-12-08 00:00:00

Problem up13-2: mz13-2 (дореш)

Процесс-родитель создает процесса-сына, а тот в свою очередь процесса-внука. Процесс-родитель и процесс-внук должны быть соединены анонимным каналом в направлении от родителя к внуку.

Процесс-родитель считывает 32-битные знаковые целые числа, подаваемые на стандартном потоке ввода в текстовом виде. Последовательность заканчивается признаком конца файла. Процесс-родитель передает считанные числа в канал в бинарном виде.

Процесс-внук считывает числа из канала и вычисляет их сумму. После чтения всех чисел процесс-внук выводит на стандартный поток вывода их сумму и завершает работу.

Процесс-родитель должен дожидаться завершения всех порожденных им процессов и завершиться сам с кодом завершения 0.

Ошибка "Synchronization error" означает, что родитель завершился раньше какого-либо из порожденных им процессов.

Examples

Input

1 2 3 4 5 6 7 8 9 10

Output

55

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...

up13-2

up14-4

mz15-1

mz15-2...

mz15-3...

mz15-4...

mz15-5...



12:22:55 / RUNNING

Submit a solution for up13-3-mz13-3 (дореш)

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	2 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022-11-24 00:00:00
Deadline:	2022-12-15 00:00:00
Date penalty formula:	2022/12/31 0-50/7d
Current penalty:	50
Next soft deadline:	2022-12-08 00:00:00

Problem up13-3: mz13-3 (дореш)

Программе в аргументах командной строки передаются три команды для выполнения, затем два имени файла (всего 5 аргументов командной строки). Программа должна запустить на выполнение процессы, чтобы они выполнялись в конфигурации, соответствующей команде shell:

```
( cmd1 < file1 && cmd2 ) | cmd3 >> file2
```

Родительский процесс должен дожидаться завершения всех созданных им процессов и сам завершить выполнение.

Считайте, что команды cmd1, cmd2, cmd3 можно запускать на выполнение с помощью exeslp.

Запись CMD1 && CMD2 означает, что процесс CMD2 должен быть запущен только после успешного завершения процесса CMD1.

Запись (CMD) используется для группировки, при этом вся командная строка CMD выполняется в отдельном процессе, порожденном основным процессом.

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

up13-3

up14-4

mz15-1

mz15-2

mz15-3

mz15-4

mz15-5



12:23:45 / RUNNING

Submit a solution for up13-4-mz13-4 (дореш)

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	1 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022-11-24 00:00:00
Deadline:	2022-12-15 00:00:00
Date penalty formula:	2022/12/31 0-50/7d
Current penalty:	50
Next soft deadline:	2022-12-08 00:00:00

Problem up13-4: mz13-4 (дореш)

Программе в аргументах командной строки передается произвольное (возможно пустое) количество команд для выполнения. Программа должна запустить на выполнение процессы, чтобы они выполнялись в конфигурации, соответствующей команде shell:

```
cmd1 | cmd2 | ... | cmdN
```

Родительский процесс должен дожждаться завершения всех созданных им процессов и сам завершить выполнение. Если все системные вызовы, исполнявшиеся родителем (например, создание процесса), завершились успешно, родитель завершается с кодом 0. В случае ошибки выполнения системного вызова (например, невозможности создания очередного процесса), родитель уничтожает уже созданные к этому моменту процессы с помощью сигнала SIGKILL и сам завершается с кодом 1 после завершения сыновей.

Команды cmd1, cmd2, ... запускайте на выполнение с помощью `exec1p`.

В родителе одновременно может быть открыто не более 6 файловых дескрипторов.

Создаваемые процессы должны оставаться в группе процессов родителя.

Программа не должна ничего выводить на стандартный поток ошибок.

Например, если скомпилированная программа называется `solution`, то ее запуск с аргументами

```
./solution ls cat wc
```

должен дать такой же результат на стандартном потоке вывода, как выполнение команды `shell`

```
ls | cat | wc
```

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
4961	2022-12-01 00:01:17	1743	up13-4	gcc-32	Pending review	14	30=100-2*10-50	View	View
4958	2022-11-30 23:57:51	1707	up13-4	gcc-32	Partial solution	13	0=0-1*10	View	View
4957	2022-11-30 23:57:23	1707	up13-4	gcc-32	Partial solution	13	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...

up13-4

up14-4

mz15-1...

mz15-2...

mz15-3...

mz15-4...

mz15-5...



12:24:11 / RUNNING

Submit a solution for up13-5-mz13-5 (дореш)

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022-11-24 00:00:00
Deadline:	2022-12-15 00:00:00
Date penalty formula:	2022/12/31 0-50/7d
Current penalty:	50
Next soft deadline:	2022-12-08 00:00:00

Problem up13-5: mz13-5 (дореш)

Родитель создает двух сыновей, связывая их двумя каналами. Сыновья начинают обмениваться числами 1, 2, 3, ... Число 1 получает первый сын. При получении числа из входного канала каждый сын печатает на стандартный поток вывода родителя свой номер (1 или 2) и полученное число и пересылает обратно число, на 1 большее. При достижении числа, задаваемого в командной строке, сын не пересылает число в канал, а завершает работу.

Процесс может завершить работу либо при получении максимального числа из канала, либо при наступлении конца файла в канале. Завершать процесс в других случаях не разрешается.

Отец дожидается завершения работы обоих процессов, затем выводит строку Done и завершает работу.

Если в командной строке передано число 5, pid первого процесса - 100, pid второго процесса - 101, то вывод должен быть таким:

```
1 1
2 2
1 3
2 4
Done
```

Данные передавать в pipe в текстовом виде, для ввода-вывода использовать высокоуровневый ввод-вывод (printf, scanf и т. д.).

Привязывать файловые дескрипторы к дескрипторам потока можно либо с помощью функции fdopen, либо с помощью перенаправления стандартных потоков ввода-вывода.

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
4924	2022-11-30 23:32:50	1876	up13-5	gcc-32	Pending review	7	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...

up13-5

up14-4

mz15-1

mz15-2...

mz15-3...

mz15-4...

mz15-5...