



22:24:48 / RUNNING

Submit a solution for ht10-1-ДЗ 10-1

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/02 09:00:00
Deadline:	2022/12/30 09:00:00
Date penalty formula:	2022/12/31 0-50/14d
Next soft deadline:	2022/11/16 09:00:00

Problem ht10-1: ДЗ 10-1

Рассмотрим модельный 16-битный процессор с 16-битным виртуальным и физическим адресным пространством, адресуемым побайтно, в котором реализована одноуровневая виртуальная страничная адресация с размером страницы 512 байт. Процессор хранит целые числа в формате Little-Endian.

Первым аргументом командной строки передается бинарный файл размером 64KiB - содержимое физической памяти. Вторым аргументом командной строки передается физический адрес начала таблицы страниц в шестнадцатеричном виде.

Таблица страниц размещается в физической памяти, начиная с указанного адреса. Каждая запись таблицы страниц имеет размер 16 бит, в которой старшие биты хранят номер физической страницы, а младшие биты равны 0.

На стандартном потоке ввода задаются выровненные виртуальные адреса 16-битных ячеек памяти в шестнадцатеричной записи. На стандартный поток вывода напечатайте значения, хранящиеся в ячейках. Значения выводите в беззнаковом десятичном виде.

Для работы с файлом с содержимым физической памяти используйте mmap.

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



22:25:39 / RUNNING

Submit a solution for ht10-2-Д3 10-2

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/02 09:00:00
Deadline:	2022/12/30 09:00:00
Date penalty formula:	2022/12/31 0-50/14d
Next soft deadline:	2022/11/16 09:00:00

Problem ht10-2: Д3 10-2

Определите структуру RandomOperations с полями destroy и next — указателями на функции.

Определите структуру RandomGenerator с необходимыми полями.

```
typedef struct RandomGenerator
{
    //...
} RandomGenerator;
```

напишите реализацию функции

```
RandomGenerator *random_create(int seed);
```

напишите реализации недостающих функций таким образом, чтобы фрагмент

```
RandomGenerator *rr = random_create(1234);
for (int j = 0; j < 100; ++j) {
    printf("%d\n", rr->ops->next(rr));
}
rr->ops->destroy(rr);
```

выводил на печать 100 псевдослучайных чисел, полученных с помощью линейного конгруэнтного генератора (Linear congruential generator) с начальным значением 1234, мультипликатором (a) 1103515245, инкрементом (c) 12345 и модулем 2^31. Затем вызов destroy освобождает память, выделенную функцией random_create.

Функция next должна возвращать *следующий* элемент псевдослучайной последовательности относительно текущего состояния.

Сдаваемый на проверку фрагмент программы должен содержать определение структур RandomOperations, RandomGenerator определение функции random_create и других функций, кроме main.

Указание. Структуру с указателями на функции можно сделать глобальной константой и использовать ее для всех создаваемых объектов.

Submit a solution

Language:gcc-vg - GNU C (valgrind) 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



22:25:53 / RUNNING

Submit a solution for ht10-3-ДЗ 10-3

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/02 09:00:00
Deadline:	2022/12/30 09:00:00
Date penalty formula:	2022/12/31 0-50/14d
Next soft deadline:	2022/11/16 09:00:00

Problem ht10-3: ДЗ 10-3

Программе в аргументах командной строки задаются имя выходного файла, число строк (rows) и число столбцов (cols).

Выходной файл - это бинарный файл, в котором записана матрица (двухмерный массив) размера rows строк на cols столбцов 32-битных целых чисел. Матрица хранится по строкам (как в Си). Запишите в выходной файл числа от 1 до rows*cols по спирали как показано на примере ниже.

Параметры rows и cols больше нуля. Матрица полностью помещается в память.

Используйте mmap для формирования выходного файла.

Например, если rows == 4, cols == 5, то в выходной файл должна быть записана матрица:

1	2	3	4	5
14	15	16	17	6
13	20	19	18	7
12	11	10	9	8

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



22:26:07 / RUNNING

Submit a solution for ht10-4-ДЗ 10-4

Full score:	100
Run penalty:	10
Time limit:	1 s
Real time limit:	5 s
Memory limit:	64M
Stack limit:	8M
Open date:	2022/11/02 09:00:00
Deadline:	2022/12/30 09:00:00
Date penalty formula:	2022/12/31 0-50/14d
Next soft deadline:	2022/11/16 09:00:00

Problem ht10-4: ДЗ 10-4

Программе в аргументах командной строки передаются следующие параметры:

- Имя динамической библиотеки для загрузки
- Имя функции в этой библиотеке
- Сигнатура функции
- Далее идут аргументы функции (могут отсутствовать)

Ваша программа должна вызвать указанную функцию, передать ей указанные параметры в соответствии с сигнатурой и вывести на стандартный поток вывода результат, возвращенный этой функцией.

Сигнатура функции — это строка, задающая тип возвращаемого значения и типы аргументов. Первый символ строки задает тип возвращаемого значения: *v* — void, *i* — int, *d* — double, *s* — char * (строка). Последующие символы строки задают типы параметров: *i* — int, *d* — double, *s* — char * (строка).

Например, сигнатура *vssid* соответствует функции вида:

```
void func(char *, char *, int, double);
```

Общий размер параметров функции не превышает 64 байта.

Если функция возвращает значение типа double, то он должен быть напечатан с помощью форматного преобразования *%.10g*. Если функция возвращает void, ничего выводить не нужно. Не забывайте выводить *\n* в конце (кроме случая void).

Пример аргументов командной строки:

```
./solution libc.so.6 printf vssid 'Hello: %s, %d, %f' abc 10 12.4
```

Результат работы:

```
Hello: abc, 10, 12.400000
```

Ваше решение будет тестироваться на платформе Linux-i386. Кросс-платформенность не требуется.

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



Submit a solution for ht10-5-Д3 10-5

Open date:	2022/11/08 19:00:00
Deadline:	2022/12/30 09:00:00
Date penalty formula:	2022/12/31 0-50/14d
Next soft deadline:	2022/11/22 19:00:00

Problem ht10-5: Д3 10-5

Дана задача. На стандартном потоке ввода подаются четыре 32-битных знаковых целых чисел L1, H1, L2, H2. Числа L1, H1 задают отрезок на числовой прямой вида [L1;H1) (то есть L1 входит в отрезок, а H1 — не входит). Гарантируется, что L1 <= H1, если L1 == H1, такой отрезок задает пустое множество. Числа L2, H2 задают границы второго отрезка на числовой прямой аналогично первому отрезку.

На стандартный поток вывода напечатайте пару 32-битных знаковых целых чисел L3, H3, задающих отрезок на числовой прямой, который является пересечением отрезков [L1;H1) и [L2;H2). Если результатом пересечения является пустое множество, оно всегда выводится в виде "0 0".

Ваша задача — написать тесты для проверки решений этой задачи. Множество тестов должно быть корректным и полным. Корректность означает, что любое правильное решение исходной задачи должно успешно проходить все тесты. Полнота означает, что любое разумно неправильное решение исходной задачи должно не проходить хотя бы один тест. Под разумно неправильным решением понимается решение, которое не выдает намеренно неправильный ответ в зависимости от случайного входного набора или для входных данных, не следующих из условия задачи. Например, "неразумное" неправильное решение может давать неправильный ответ только если значение H2 равно 1231412421. Такие "неразумные" неправильные решения рассматривать не нужно.

Тесты состоят из набора тестовых пар <входные данные;правильный ответ>. Входные данные должны быть корректными и удовлетворять ограничениям задачи. В данном случае, входные данные должны представлять собой четыре целых числа, представимых 32-битным знаковым типом, с указанными ограничениями на границы. Числа могут разделяться пробельными символами произвольным образом. Правильный ответ должен быть корректен. В данном случае, правильный ответ должен представлять собой два числа, являющихся правильным ответом на входные данные. Числа могут разделяться пробельными символами произвольным образом.

Файлы с входными данными должны называться 001.dat, 002.dat, 003.dat и т. д. Соответствующие файлы с правильным ответом должны называться 001.ans, 002.ans, 003.ans и т. д. Размер любого файла не должен превышать 1KiB.

Тестовый набор необходимо сдать на проверку в виде архива в формате .tgz. Архив должен содержать единственный каталог с именем tests, в котором должны размещаться разработанные вами тесты. Количество тестов не должно превышаться 20.

Для создания архива формата .tgz в командной строке выполните команду:

```
tar cfz arch.tgz tests
```

Submit a solution

File Файл не выбран

Send!

Previous submissions of this problem (last 15)

Run ID	Time	Size	Problem	Language	Result	Tests passed	Score	View source	View report
3959	2022/11/12 13:17:18	429	ht10-5	N/A	OK	0	100	View	View
3958	2022/11/12 13:05:25	387	ht10-5	N/A	Partial solution	0	0	View	View
3957	2022/11/12 13:04:39	377	ht10-5	N/A	Partial solution	0	0	View	View
3956	2022/11/12 13:03:42	373	ht10-5	N/A	Partial solution	0	0	View	View
3955	2022/11/12 13:02:49	382	ht10-5	N/A	Partial solution	0	0	View	View
3954	2022/11/12 13:00:40	323	ht10-5	N/A	Partial solution	0	0	View	View
3953	2022/11/12 13:00:08	320	ht10-5	N/A	Partial solution	0	0	View	View
3952	2022/11/12 12:58:15	284	ht10-5	N/A	Partial solution	0	0	View	View
3951	2022/11/12 12:56:22	285	ht10-5	N/A	Presentation error	0	0	View	View
3950	2022/11/12 12:54:07	238	ht10-5	N/A	Partial solution	0	0	View	View
3949	2022/11/12 12:52:41	532	ht10-5	N/A	Coding violation style	N/A	N/A	View	View
3948	2022/11/12 12:50:02	199	ht10-5	N/A	Partial solution	0	0	View	View
3947	2022/11/12 11:56:43	849	ht10-5	N/A	Partial solution	0	0	View	View
3946	2022/11/12 11:55:11	843	ht10-5	N/A	Partial solution	0	0	View	View
3944	2022/11/12 11:47:35	728	ht10-5	N/A	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

ht10-5