

Справочная информация:

Типы данных в C. Приведены размеры для 64-х битных систем.

Спецификатор	Тип	Размер (байт)	Спецификатор	Тип	Размер (байт)
%h	short	2	%c	char	1
%uh	unsigned short	2	%c	unsigned char	1
%d или %i	int	4	%f	float	4
%u	unsigned int	4	%lf	double	8
%ld	long	8 (или 4)	%Lf	long double	обычно 10
%lu	unsigned long	8 (или 4)	%p	указатель (<имя типа>*)	8
%lld	long long	8	%s	Строка	
%llu	unsigned long long	8			

Многие функции в языке C возвращают особый тип `size_t`. Часто это просто `unsigned long`:

```
typedef unsigned long size_t;
```

Задачи:

Функции не должны ничего считывать и печатать.

1. **Остаток:** Написать программу, которая считывает 2 числа a и b и печатает их остаток. $0 \leq a, b \leq 2^{32} - 1$. Использовать тип `unsigned int`.
2. **Произведение чисел:** Написать программу, которая считывает 2 числа a и b и печатает их произведение. $0 \leq a, b \leq 2^{32} - 1$.
3. **mod 7:** Написать программу, которая печатает все числа делящиеся на 7 в интервале от 700 до 1000, используя цикл `for`.
4. **Часть года:** Написать функцию на вход которой подаётся целое число – число дней прошедших с начала года. Она должна возвращать вещественное число типа `float` – доля прошедшего года (от 0 до 1). В году 365 дней.
5. **Математическая функция:** Написать функцию, которая вычисляет выражение $\sin(\sqrt{|x|})$. Использовать числа двойной точности `double`. Функция для вычисления модуля вещественного числа – `fabs()` из библиотеки `math.h`.
6. **Нормализация:** На вход программе подаётся целое число n и n вещественных чисел типа `float`. Нужно эти числа нормировать (то есть разделить на их сумму) и напечатать.
7. **Sqared Matrix:** Написать функцию `void matrix_square(int n, int arr[SIZE][SIZE])`, которая возводит двумерную матрицу в квадрат. `SIZE` – максимально возможный размер массива, задаётся так:

```
#define SIZE 100
```