

# Задача A. A xor B =

Автор: M. Liamaev, D. Glushkova

Ограничение времени: 10 сек  
Ограничение памяти: 256 Мб

**Условие**

Реализуйте в программе *Logisim – evolution* функцию  $A \text{ xor } B = C$ .

**Формат входного файла**

Тестируемая схема должна называться `main`. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке *PE*.

Входы		
Label/Метка	Битность	Описание
A	1	Вход A
B	1	Вход B

Выходы		
Label/Метка	Битность	Описание
C	1	Результат $A \text{ xor } B$

**Примеры тестов**

№	Входной файл (*NONE)	Выходной файл (*NONE)
1	A B C 0 0 0 1 0 1	Тест запускается через GUI Logisim Simulate → Test Vector

A

B

C

D

E

F

G

H

I

K

L

M

N

---

# Задача В. Полусумматор 1b

Автор: М. Liamaev, D. Glushkova

Входной файл: Стандартный вход

Выходной файл: Стандартный выход

Ограничение времени: 10 сек

Ограничение памяти: 256 Мб

## Условие

Реализуйте в программе *Logisim – evolution* полусумматор 1bit.

## Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке *PE*.

Использование сумматора из стандартной библиотеки *Logisim* запрещено. Нарушение этого требования приведет к ошибке *LI*.

A
B
C
D
E
F
G
H
I
K
L
M
N

Входы		
Label/Метка	Битность	Описание
A	1	Вход A
B	1	Вход B

Выходы		
Label/Метка	Битность	Описание
S	1	Результат суммы $A + B$
P	1	Бит переноса

Примеры тестов

№	Стандартный вход	Стандартный выход
1	A B S P 0 0 0 0 1 0 1 0 0 1 1 0 1 1 0 1	Тест запускается через GUI Logisim Simulate → Test Vector



# Задача С. Сумматор 1b =

Автор: M. Liamaev, D. Glushkova

Входной файл: Стандартный вход

Выходной файл: Стандартный выход

Ограничение времени: 10 сек

Ограничение памяти: 256 Мб

## Условие

Реализуйте в программе *Logisim – evolution* сумматор 1 bit.

## Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке PE.

Использование сумматора из стандартной библиотеки *Logisim* запрещено. Нарушение этого требования приведет к ошибке LI.

A
B
C
D
E
F
G
H
I
K
L
M
N

Входы		
Label/Метка	Битность	Описание
A	1	Слагаемое A
B	1	Слагаемое B
P0	1	Входной бит переноса

Выходы		
Label/Метка	Битность	Описание
S	1	Результат суммы $A + B$
P	1	Выходной бит переноса

Примеры тестов

№	Стандартный вход	Стандартный выход
1	P0 A B S P 0 0 0 0 0 0 0 1 1 0 0 1 0 1 0 0 1 1 0 1 1 0 0 1 0 1 0 1 0 1 1 1 0 0 1 1 1 1 1 1	Тест запускается через GUI Logisim Simulate → Test Vector

# Задача D. Сумматор 8b =

Автор: M. Liamaev, D. Glushkova

Входной файл: Стандартный вход

Выходной файл: Стандартный выход

Ограничение времени: 10 сек

Ограничение памяти: 256 Мб

Условие

Реализуйте в программе *Logisim – evolution* сумматор 8 bit.

Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке *PE*.

Использование сумматора из стандартной библиотеки *Logisim* запрещено. Нарушение этого требования приведет к ошибке *LI*.

A

B

C

D

E

F

G

H

I

K

L

M

N

Входы			
Label/Метка	Битность	Тип	Описание
A	8	Беззнаковый	Вход A
B	8	Беззнаковый	Вход B

Выходы			
Label/Метка	Битность	Тип	Описание
S	8	Беззнаковый	Результат $A + B$

Примеры тестов

№	Стандартный вход	Стандартный выход
1	A[8] B[8] S[8] 3 3 6 141 100 241 141 101 242 141 110 251 141 111 252 141 112 253 141 113 254 141 114 255 141 115 0 141 116 1 141 117 2 141 123 8 141 124 9 141 125 10	Тест запускается через GUI Logisim Simulate → Test Vector



# Задача Е. Вычитатель 8b =

Автор: М. Liamaev, D. Glushkova

Входной файл: Стандартный вход

Выходной файл: Стандартный выход

Ограничение времени: 20 сек

Ограничение памяти: 256 Мб

Условие

Реализуйте в программе *Logisim – evolution* вычитатель 8 bit для знаковых чисел.

Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке PE.

Использование сумматора и вычитателя из стандартной библиотеки *Logisim* запрещено. Нарушение этого требования приведет к ошибке LI.

A

B

C

D

E

F

G

H

I

K

L

M

N

Входы			
Label/Метка	Битность	Тип	Описание
A	8	Знаковое в дополнительном коде	Вход A
B	8	Знаковое в дополнительном коде	Вход B

Выходы			
Label/Метка	Битность	Тип	Описание
C	8	Знаковое в дополнительном коде	Результат $A - B$

Примеры тестов

№	Стандартный вход	Стандартный выход
1	A[8] B[8] C[8] 0x2 0x1 0x1 0x1 0x2 0xff 0xff 0x2 0xfd 0xff 0xfd 0x2 0x81 0xa5 0xdc 0x1c 0xd1 0x4b 0x1c 0xd9 0x43 0x1d 0xed 0x30 0x1d 0x5 0x18 0x1d 0x6c 0xb1	Тест запускается через GUI Logisim Simulate → Test Vector

# Задача F. Мультиплексор

Автор: M. Liamaev, D. Glushkova

Входной файл: Стандартный вход

Выходной файл: Стандартный выход

Ограничение времени: 10 сек

Ограничение памяти: 256 Мб

Условие

Реализуйте в программе *Logisim – evolution* мультиплексор 4 bit 8 в 1.

Таблица выбора	
Значение S	Действие
000	B = A0
001	B = A1
010	B = A2
011	B = A3
100	B = A4
101	B = A5
110	B = A6
111	B = A7

Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке *PE*.  
Использование мультиплексора из стандартной библиотеки *Logisim* запрещено. Нарушение этого требования приведет к ошибке *LI*.

A

B

C

D

E

F

G

H

I

K

L

M

N

Входы		
Label/Метка	Битность	Описание
A0	4	Вход A0
A1	4	Вход A1
A2	4	Вход A2
A3	4	Вход A3
A4	4	Вход A4
A5	4	Вход A5
A6	4	Вход A6
A7	4	Вход A7
S	3	Вход выбора

Выходы		
Label/Метка	Битность	Описание
B	4	Результат выбора

Примеры тестов

№	Стандартный вход	Стандартный выход
1	A0[4] A1[4] A2[4] A3[4] A4[4] A5[4] A6[4] A7[4] S[3] B[4] 0 2 1 5 4 7 3 6 0 0 0 2 1 5 4 7 3 6 1 2 0 2 1 5 4 7 3 6 2 1 0 2 1 5 4 7 3 6 3 5 0 2 1 5 4 7 3 6 4 4 0 2 1 5 4 7 3 6 5 7 0 2 1 5 4 7 3 6 6 3 0 2 1 5 4 7 3 6 7 6	Тест запускается через GUI Logisim Simulate → Test Vector

# Задача G. Битовые сдвиги

Автор: M. Liamaev, D. Glushkova

Входной файл: Стандартный вход

Выходной файл: Стандартный выход

Ограничение времени: 10 сек

Ограничение памяти: 256 Мб

Условие

Реализуйте в программе *Logisim – evolution* логический, арифметический и циклический сдвиги на заданное количество разрядов.

Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке *PE*.

Запрещено использовать стандартное умножение, иначе получите ошибку *LI*

A

B

C

D

E

F

G

H

I

K

L

M

N

Входы		
Label/Метка	Битность	Описание
Inp	8	Входное число
Method	2	Метод сдвига (00-логический, 01-арифметический, 10-циклический)
Shift	3	Число разрядов для сдвига
Left_Right	1	Флаг направления сдвига (влево = 0, вправо = 1)

Выходы		
Label/Метка	Битность	Описание
Ans	8	Результат сдвига

Примеры тестов

№	Стандартный вход	Стандартный выход
1	Inp[8] Method[2] Shift[3] Left_Right[1] Ans[8] 3 1 7 0 10000000 3 1 7 1 00000000 3 1 1 1 00000001 3 1 1 1 00000001 3 1 1 0 00000110 3 0 1 1 1 31 0 3 1 3 31 0 3 0 11111000	Тест запускается через GUI Logisim Simulate → Test Vector
2	Inp[8] Method[2] Shift[3] Left_Right[1] Ans[8] 12 1 1 0 24 12 1 1 1 6 12 0 1 0 24 12 0 1 1 6 12 1 3 0 96 76 0 3 1 9 76 0 3 0 96	Тест запускается через GUI Logisim Simulate → Test Vector

# Задача Н. Цифровой компаратор 8b =

Автор: М. Liamaev, D. Glushkova

Входной файл: Стандартный вход

Выходной файл: Стандартный выход

Ограничение времени: 10 сек

Ограничение памяти: 256 Мб

Условие

Реализуйте в программе *Logisim – evolution* цифровой компаратор 8 bit. Обратите внимание, что сравниваемые числа знаковые (в дополнительном коде).

Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке *PE*.

Использование компаратора из стандартной библиотеки *Logisim* запрещено. Нарушение этого требования приведет к ошибке *LI*.

A

B

C

D

E

F

G

H

I

K

L

M

N

Входы			
Label/Метка	Битность	Тип	Описание
A	8	Знаковое в дополнительном коде	Вход A
B	8	Знаковое в дополнительном коде	Вход B

Выходы		
Label/Метка	Битность	Описание
more_eq	1	1 если $A \geq B$
more	1	1 если $A > B$
equal	1	1 если $A = B$
less	1	1 если $A < B$
less_eq	1	1 если $A \leq B$

Примеры тестов

№	Стандартный вход	Стандартный выход
1	A[8] B[8] more_eq more equal less less_eq 0x48 0x47 1 1 0 0 0 0x48 0x48 1 0 1 0 1 0x48 0x49 0 0 0 1 1 0x81 0x80 1 1 0 0 0 0x81 0x81 1 0 1 0 1 0x81 0xfd 0 0 0 1 1 0x81 0xfe 0 0 0 1 1 0x81 0xff 0 0 0 1 1 0x97 0x7e 0 0 0 1 1 0x98 0x80 1 1 0 0 0 0x98 0x98 1 0 1 0 1 0x48 0x7f 0 0 0 1 1 0x49 0x80 1 1 0 0 0	Тест запускается через GUI Logisim Simulate → Test Vector



# Задача I. Умножение 8b =

Автор: М. Liamaev, D. Glushkova

Входной файл: Стандартный вход

Выходной файл: Стандартный выход

Ограничение времени: 15 сек

Ограничение памяти: 256 Мб

Условие

Реализуйте в программе *Logisim – evolution* функцию  $A * B = C$ . Умножение должно быть знаковым (используйте дополнительный код).

Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке *PE*.

Запрещено использовать стандартное умножение, иначе получите ошибку *LI*

Входы			
Label/Метка	Битность	Тип	Описание
A	8	Знаковый (дополнительный код)	Вход A
B	8	Знаковый (дополнительный код)	Вход B

Выходы			
Label/Метка	Битность	Тип	Описание
C	16	Знаковый (дополнительный код)	Результат $A * B$

Примеры тестов

№	Стандартный вход	Стандартный выход
1	A[8] B[8] C[16] 0 1 0 1 1 1	Тест запускается через GUI Logisim Simulate → Test Vector
2	A[8] B[8] C[16] 12 2 24 15 10 150	Тест запускается через GUI Logisim Simulate → Test Vector

A

B

C

D

E

F

G

H

I

K

L

M

N

---

Условие

Реализуйте в программе *Logisim – evolution* деление (/) и нахождение остатка в одной схеме для знаковых целых чисел, представленных в дополнительном коде.

Обработка отрицательных чисел должна проводится по стандартным правилам математики, а не стандартам *IEEE*. Под остатком понимается число, полученное по формуле  $D = A - B * C$

Примеры
$-7 \bmod -3 = -1$
$-7 \bmod 3 = -1$
$7 \bmod -3 = 1$
$7 \bmod 3 = 1$
$7 \bmod 0 = 7$
$7 \operatorname{div} 0 = 0$

Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке *PE*.

Запрещено использовать стандартное деление, иначе получите ошибку *LI*.

Входы		
Label/Метка	Битность	Описание
A	8	Делимое
B	8	Делитель

Выходы		
Label/Метка	Битность	Описание
C	8	Неполное частное
D	8	Остаток

Примеры тестов

№	Стандартный вход	Стандартный выход
1	A[8] B[8] C[8] D[8] 0x80 0xa 0xf4 0xf8 0x78 0x3 0x28 0x0 0x7f 0x54 0x1 0x2b	Тест запускается через GUI Logisim Simulate → Test Vector

# Задача L. Деление 8b усложненное =

Автор: M. Liamaev, D. Glushkova  
Входной файл: Стандартный вход  
Выходной файл: Стандартный выход

Ограничение времени: 15 сек  
Ограничение памяти: 256 Мб

## Условие

Реализуйте в программе *Logisim – evolution* деление (/) и нахождение остатка в одной схеме для знаковых целых чисел, представленных в дополнительном коде. Отличие этого задания от предыдущего в том, что здесь вам **запрещено пользоваться подсхемой умножения**. Обработка отрицательных чисел должна проводится по стандартным правилам математики, а не стандартам *IEEE*. Под остатком понимается число, полученное по формуле  $D = A - B * C$

Примеры
$-7 \bmod -3 = -1$
$-7 \bmod 3 = -1$
$7 \bmod -3 = 1$
$7 \bmod 3 = 1$
$7 \bmod 0 = 7$
$7 \div 0 = 0$

## Формат входных данных

Тестируемая схема должна называться main. Входы и выходы схемы должны быть названы так же, как в таблицах ниже. Нарушение этого требования приведет к ошибке *PE*.  
Запрещено использовать стандартное деление и умножение, иначе получите ошибку *LI*.

Входы		
Label/Метка	Битность	Описание
A	8	Делимое
B	8	Делитель

Выходы		
Label/Метка	Битность	Описание
C	8	Неполное частное
D	8	Остаток

## Примеры тестов

№	Стандартный вход	Стандартный выход
1	A[8] B[8] C[8] D[8] 0x80 0xa 0xf4 0xf8	Тест запускается через GUI Logisim Simulate → Test Vector

A
B
C
D
E
F
G
H
I
K
L
M
N

# Задача М. Ассемблер: корень n-ой степени =

Автор: М. Liamaev, D. Glushkova

Ограничение времени: 10 сек  
Ограничение памяти: 256 Мб

## Условие

Реализуйте программу, вычисляющую корень N-ой степени (при натуральном N), из вещественного числа  $A > 0$  с некоторой заданной точностью  $\epsilon > 0$ , воспользовавшись алгоритмом быстрого возведения в степень и методом бисекции. Если  $A < 0$  и  $N \% 2 == 0$  то выводите -1

## Отправка решения и тестирование

Для отправки задачи выберите среду разработки AnswerText и прикрепите текст программы. Данная задача будет проверяться с помощью тестов, половину из которых можете увидеть [ЗДЕСЬ](#).

## Формат входного файла

Во входной строке содержатся A, N, E — вещественное число, степень и точность соответственно.

## Формат выходного файла

Выходной файл должен содержать единственное вещественное число — корень N-ой степени для заданного числа A.

## Примеры тестов

№	Входной файл (*NONE)	Выходной файл (*NONE)
1	8 3 0.0001	2.000000
2	30 9 0.001	1.459233

A

B

C

D

E

F

G

H

I

K

L

M

N

№	Входной файл (*NONE)	Выходной файл (*NONE)
1	8 3 0.0001	2.000000
2	30 9 0.001	1.459233

# Задача N. Ассемблер: палиндромы

Автор: M. Liamaev, D. Glushkova

Ограничение времени: 10 сек  
Ограничение памяти: 256 Мб

**Условие**

Реализуйте программу, находящую для заданного набора символов все возможные палиндромы, которые можно составить из них.

**Отправка решения и тестирование**

Для отправки задачи выберите среду разработки AnswerText и прикрепите текст программы. Данная задача будет проверяться с помощью тестов, половину из которых можете увидеть [ЗДЕСЬ](#).

**Формат входного файла**

Входной файл содержит единственную строку S — набор символов.

**Формат выходного файла**

Выходной файл должен содержать все возможные палиндромы

**Примеры тестов**

№	Входной файл (*NONE)	Выходной файл (*NONE)
1	aibohpho	a b h hh hah hbh hih hoh hooh hoaoh hoboh hoioh hopoh hph i o oo oao obo oho ohho ohaho ohbho ohiho ohpho oio opo p
2	uzyw	u w y z

A  
B  
C  
D  
E  
F  
G  
H  
I  
L  
M  
N