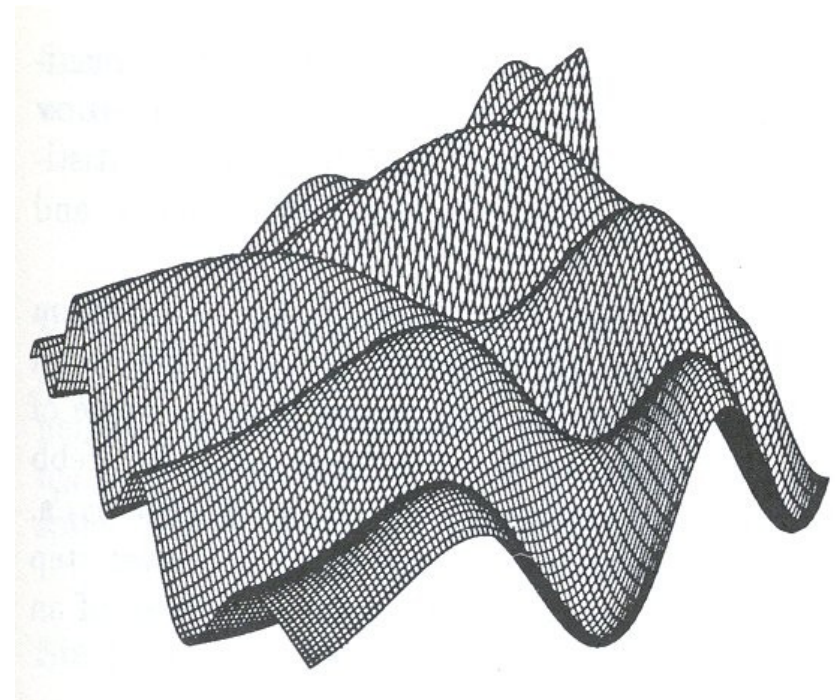
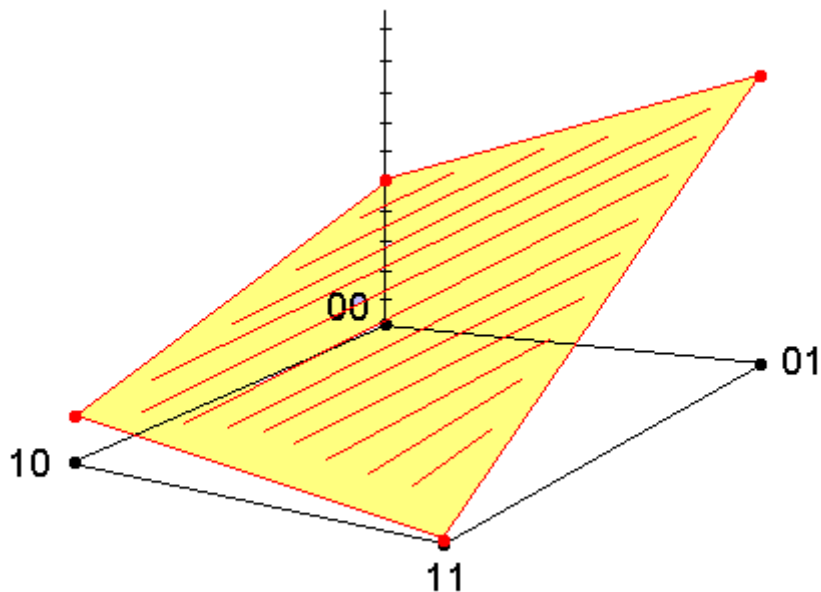


Key GA Concepts

- Search space
- Fitness landscape
- Local minimum / maximum
- Hill-climbing search
- Population-based search
- Schemas



Binary Strings of Length N

$N = 1$ 0 1

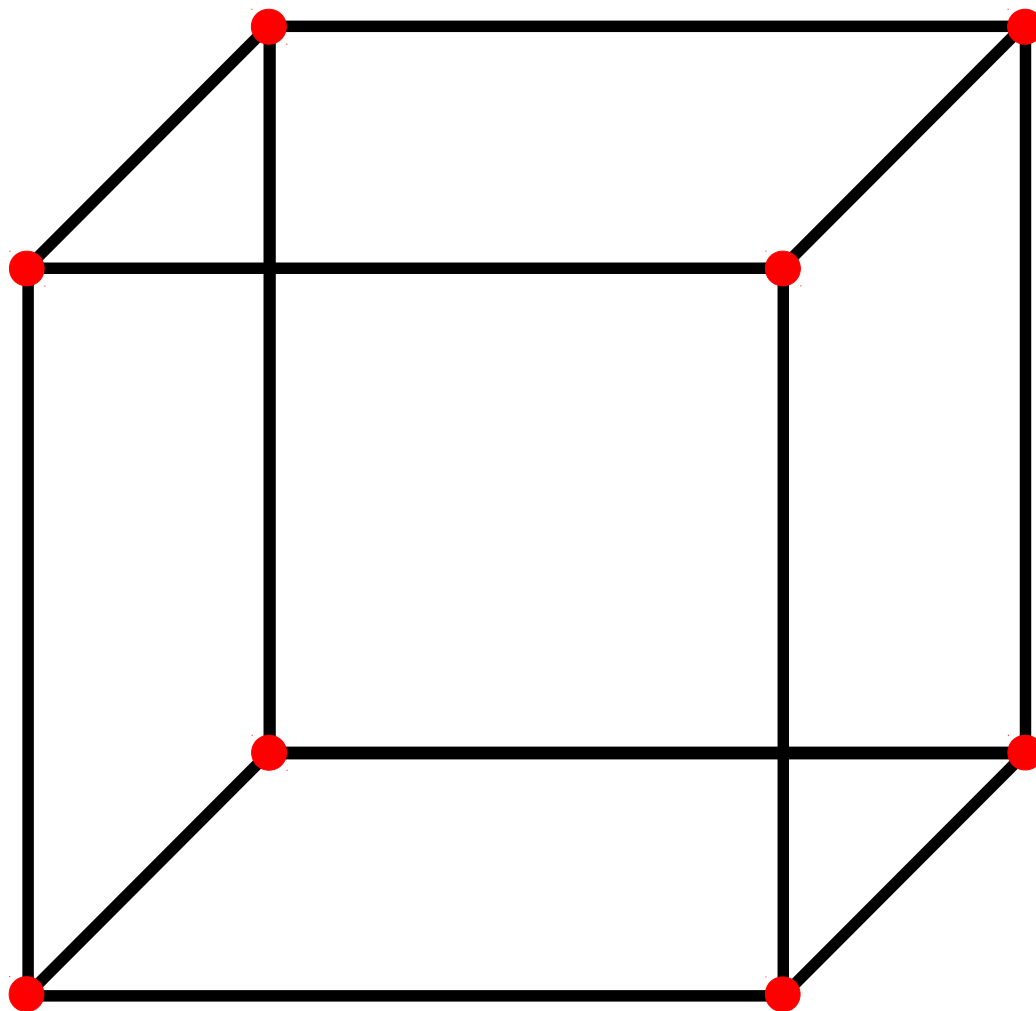
$N = 2$ 00 01 10 11

$N = 3$ 000 001 010 011 100 101 110 111

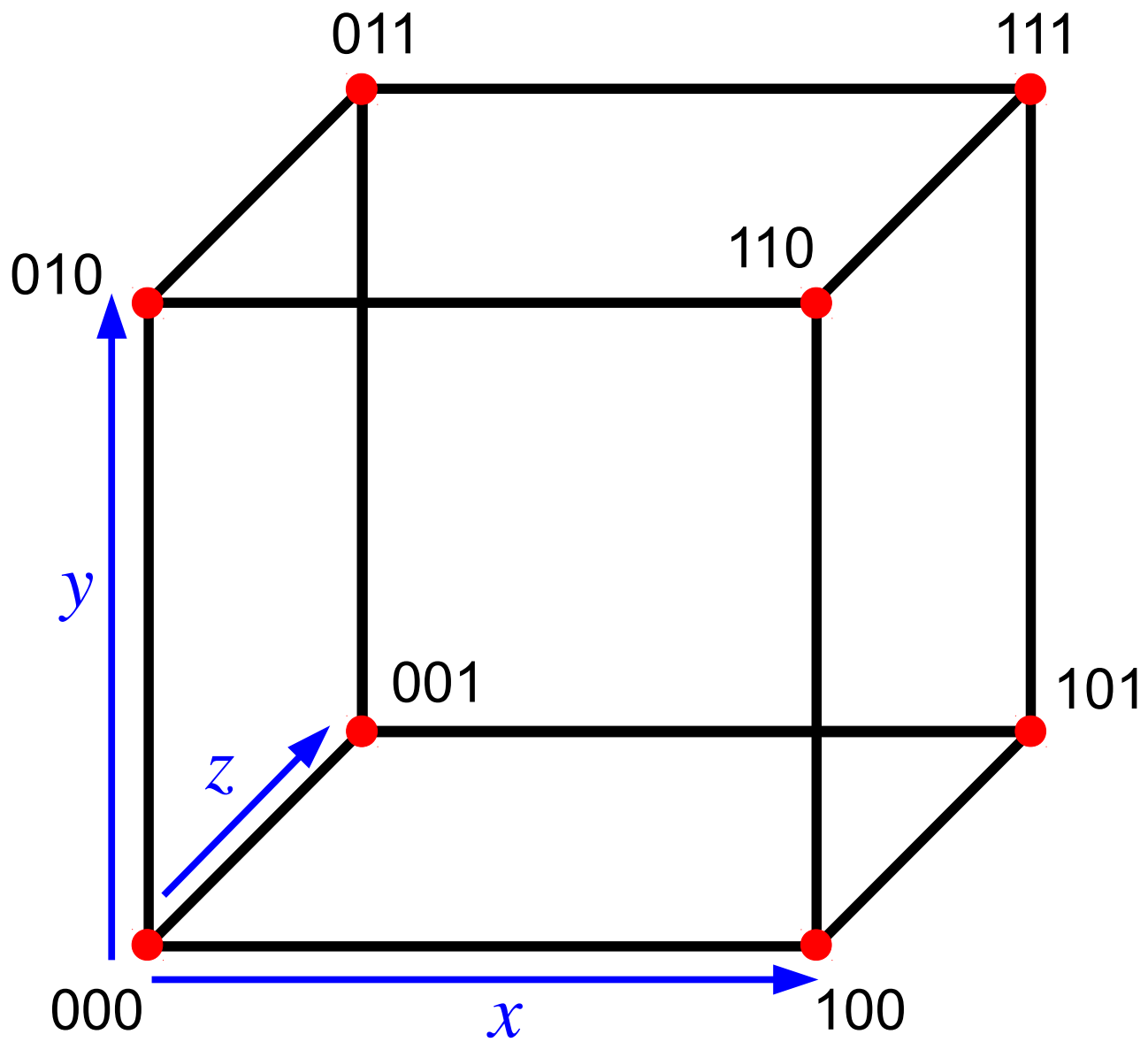
$N = 4$ 0000 0001 0010 0011 0100 0101 0110 0111
 1000 1001 1010 1011 1100 1101 1110 1111

In general, there are 2^N possible strings of length N

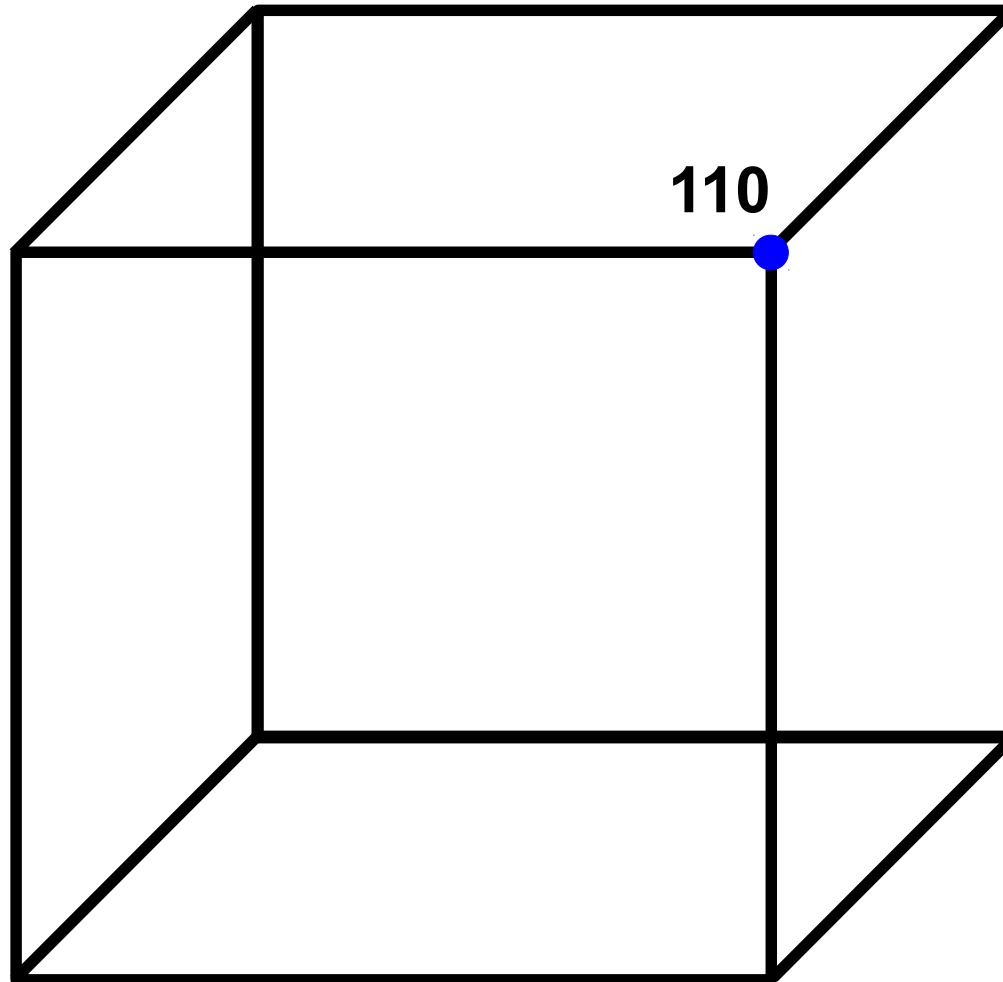
$N = 3$



xyz

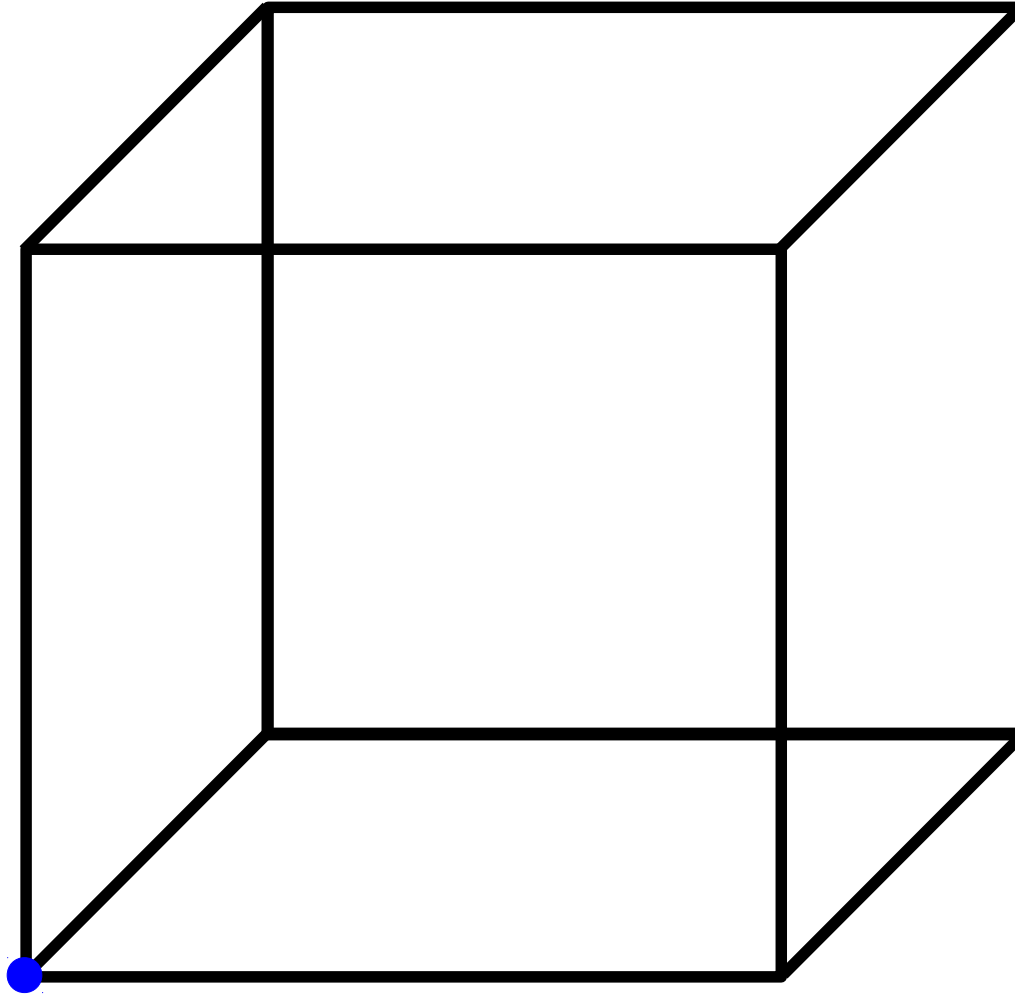


110



$2^0 = 1$ string

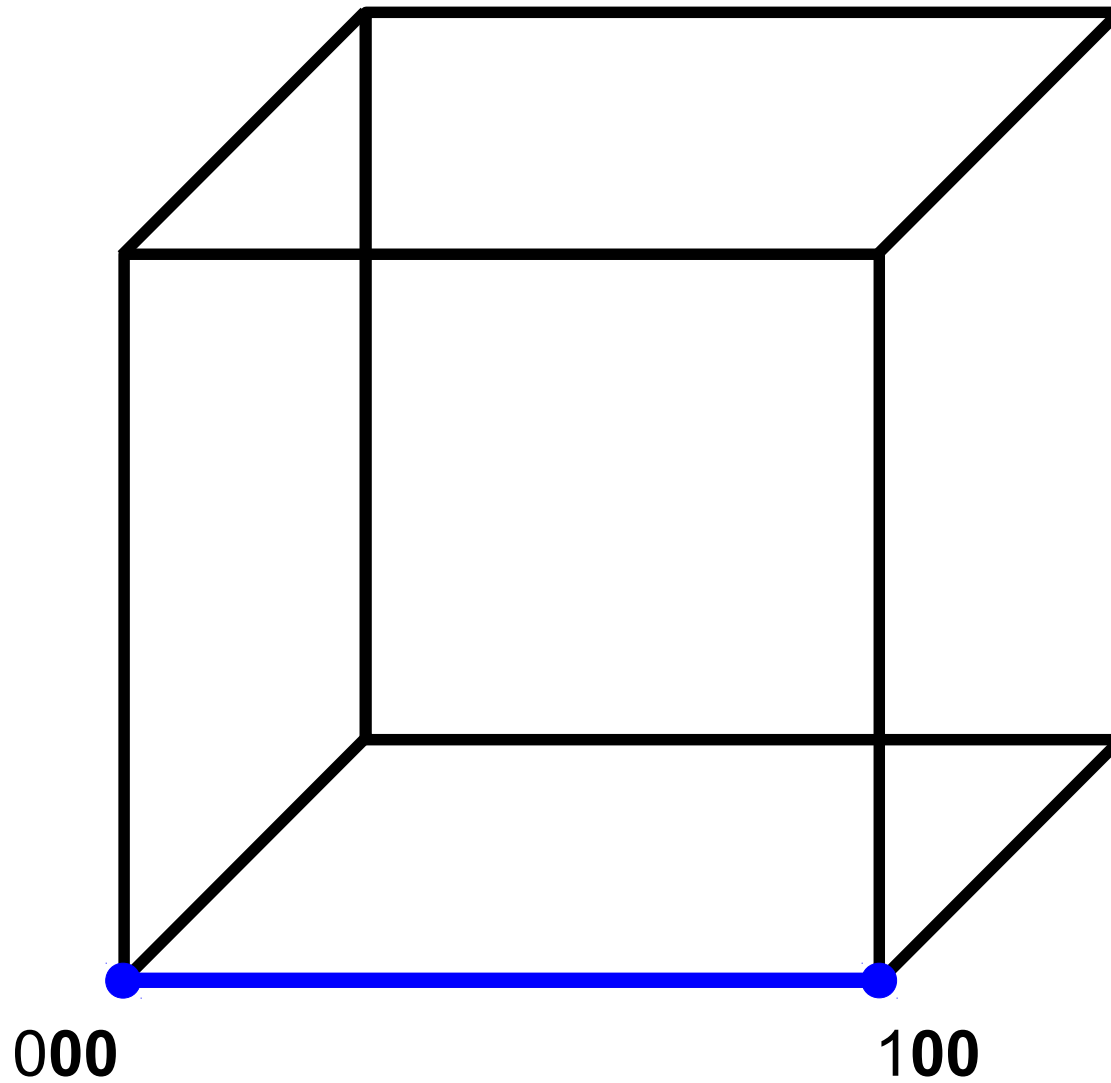
000



000

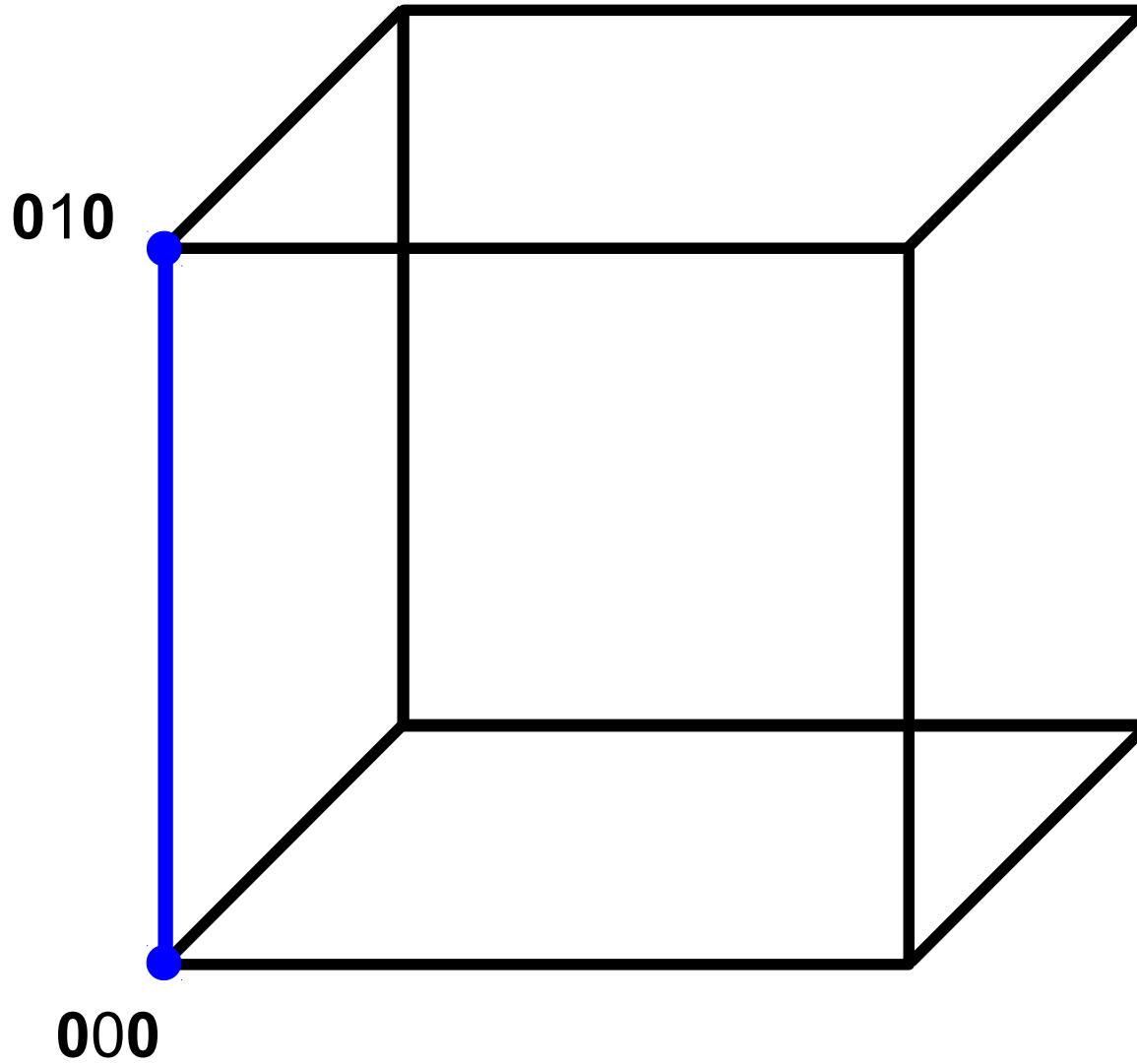
$2^0 = 1$ string

***00**



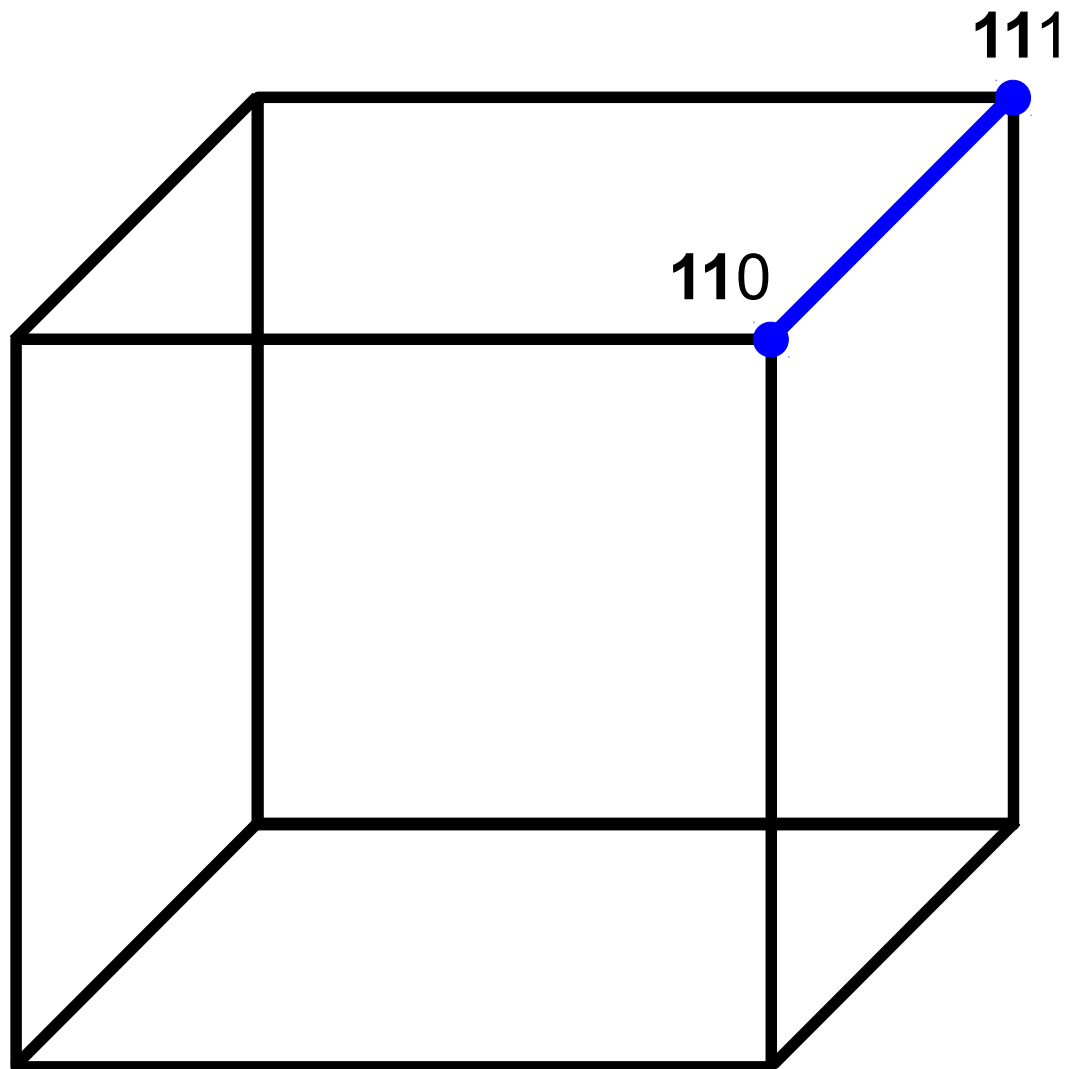
$2^1 = 2$ strings

0*0



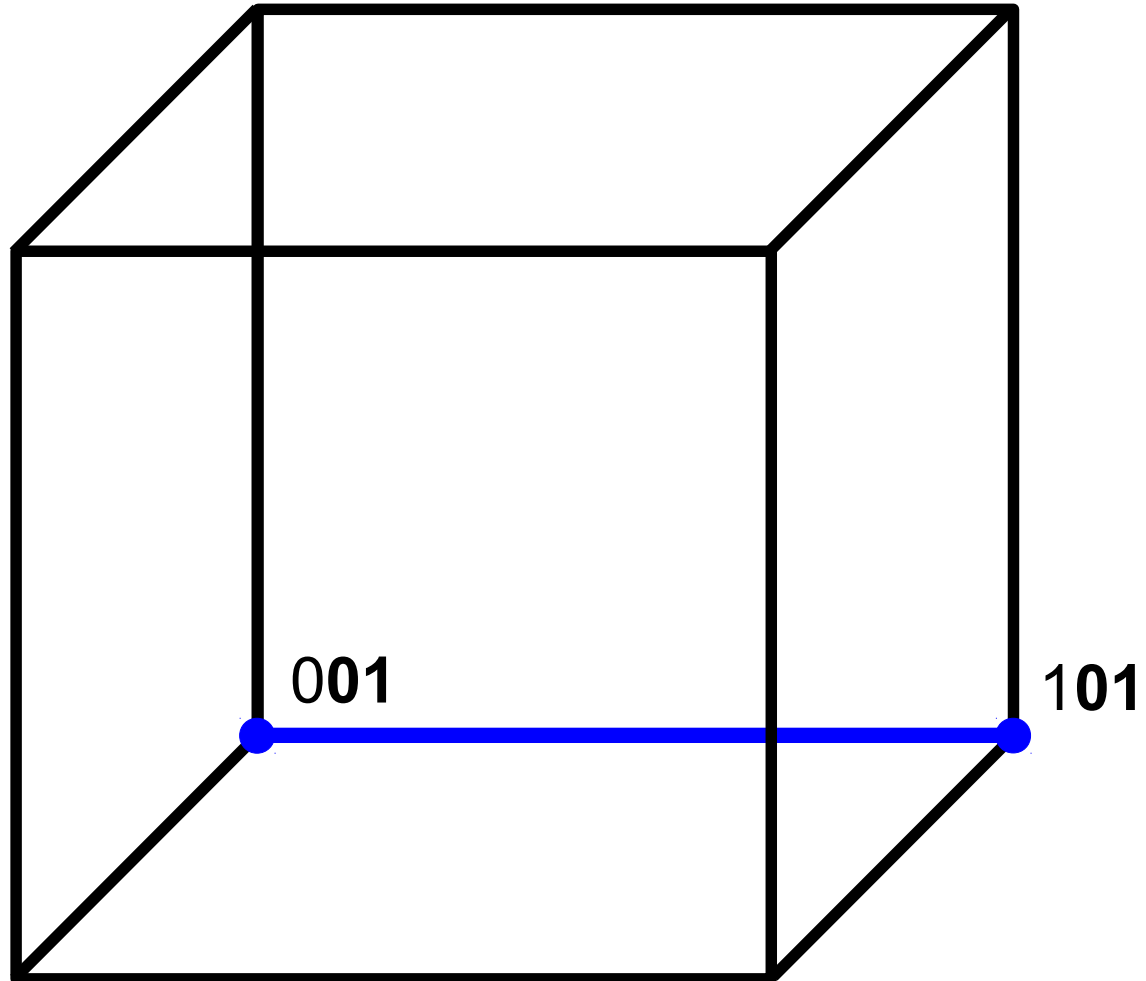
$2^1 = 2$ strings

11*



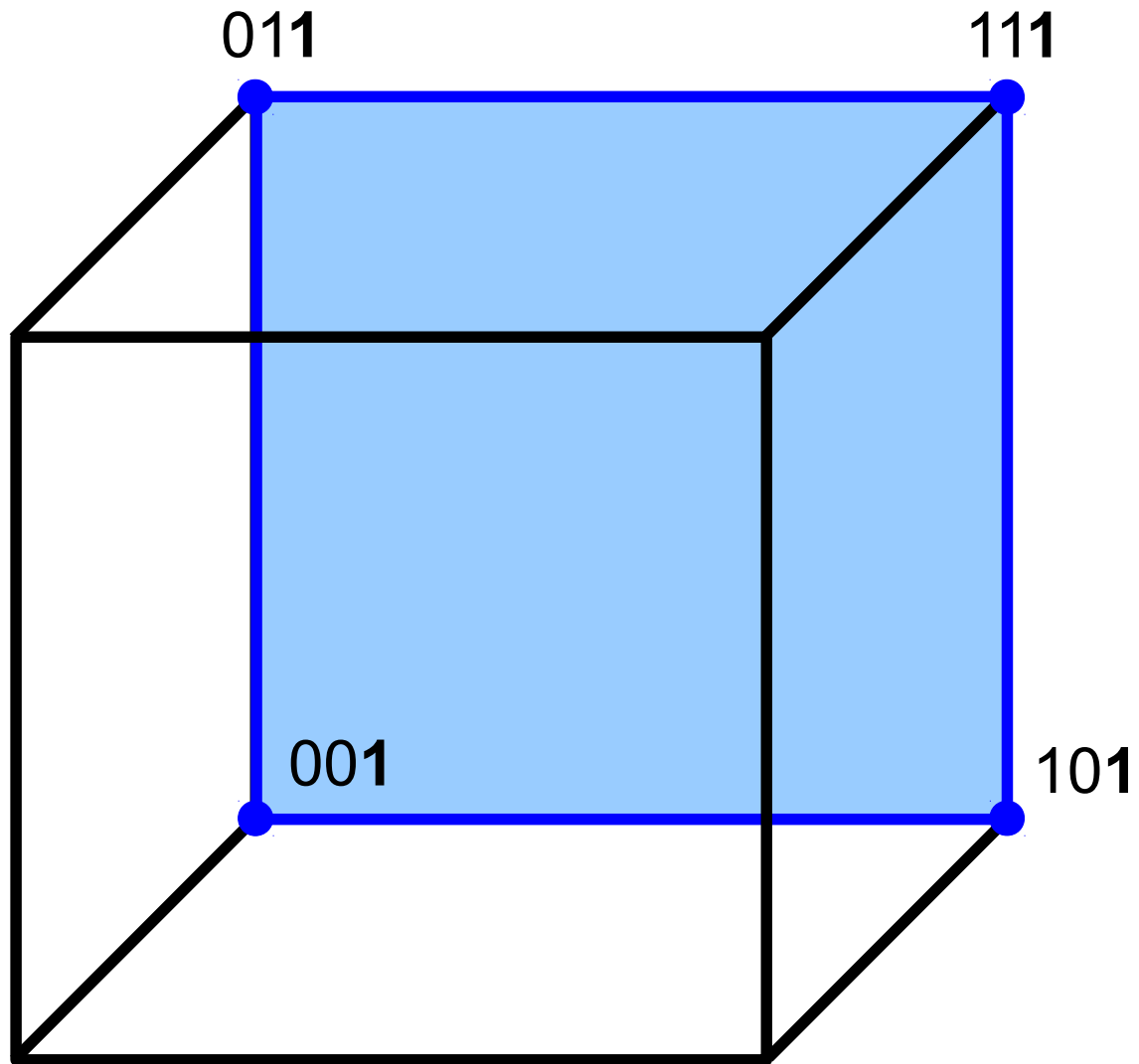
$2^1 = 2$ strings

***01**



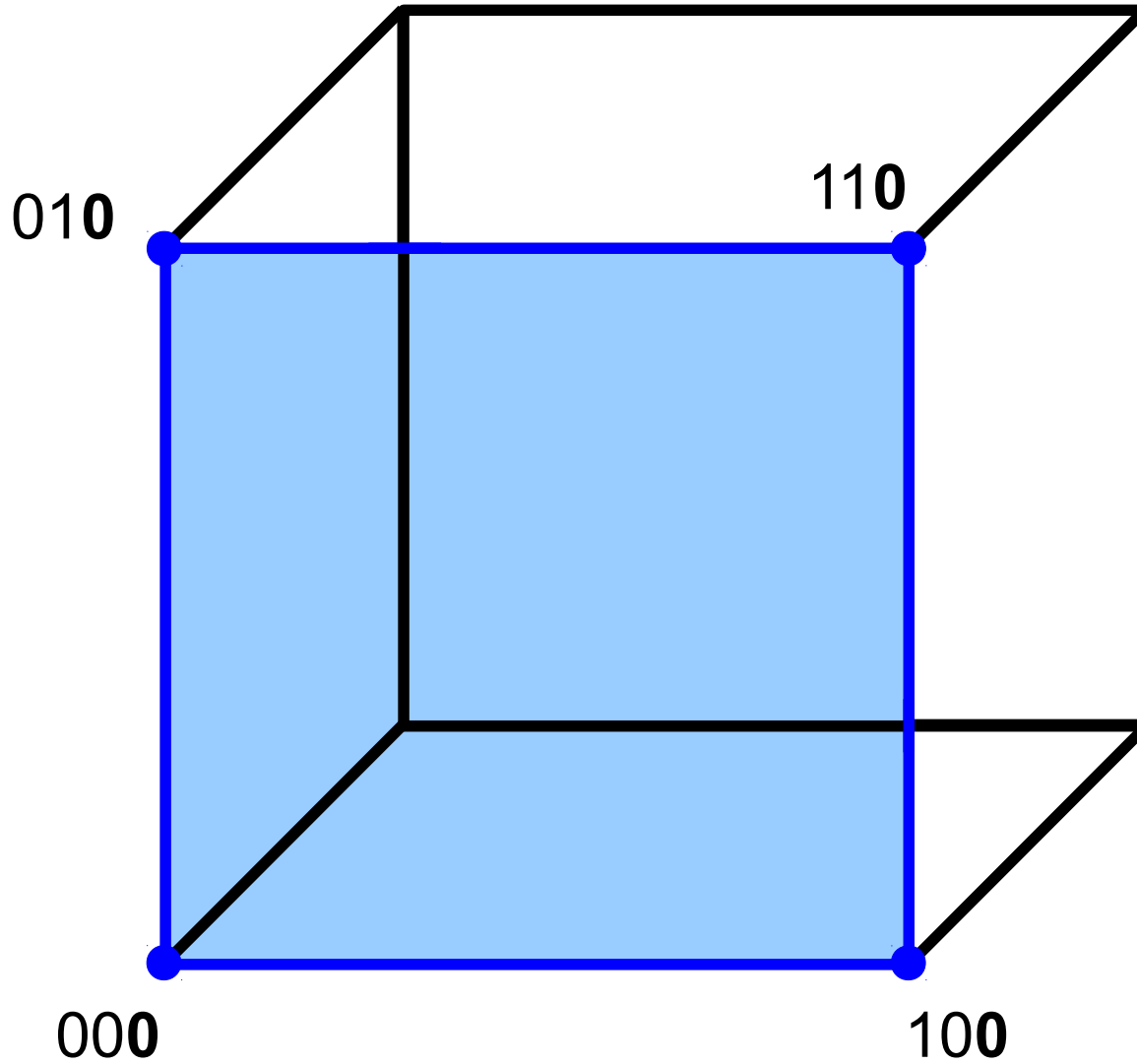
$2^1 = 2$ strings

****1**



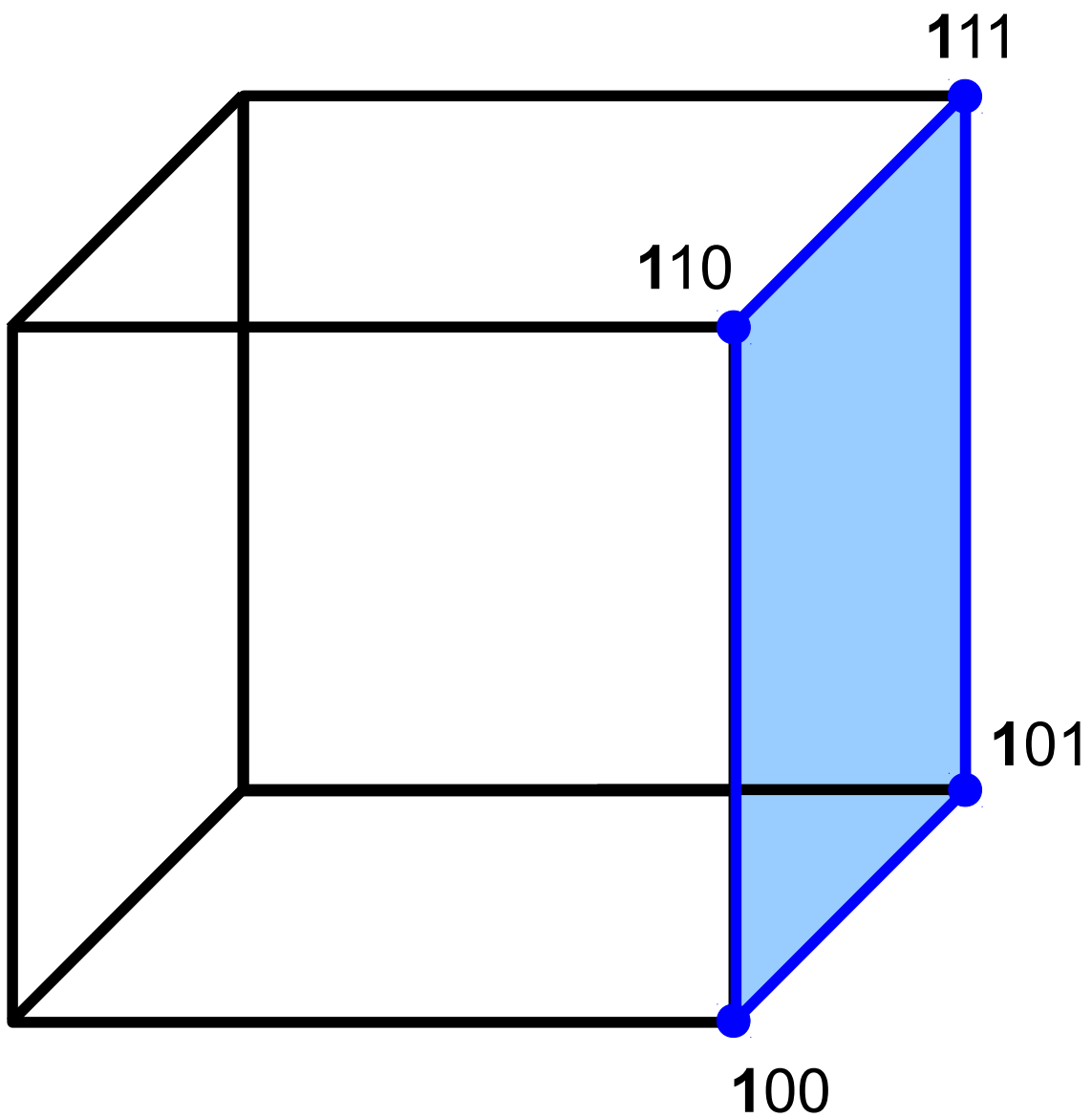
$2^2 = 4$ strings

****0**



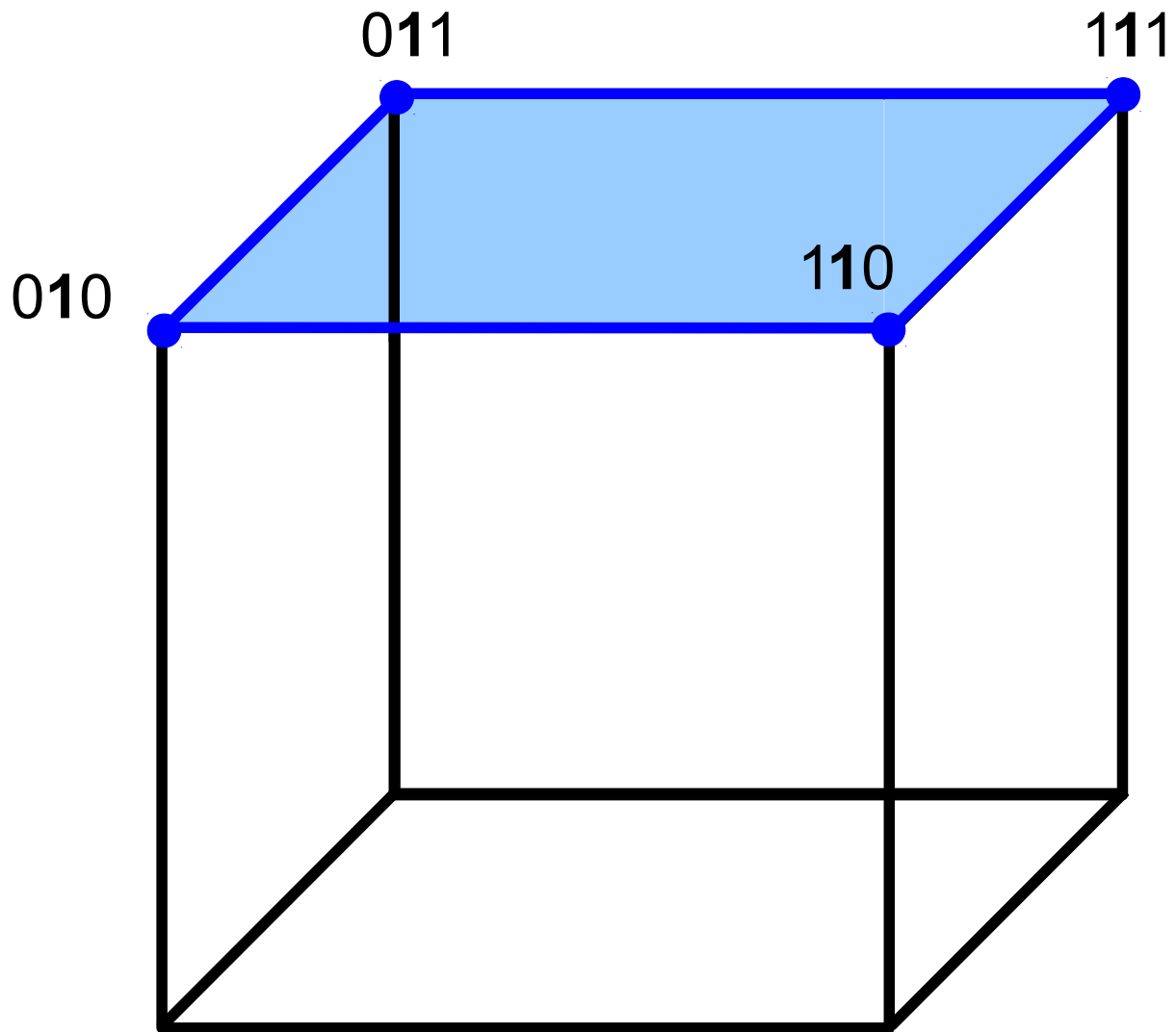
$2^2 = 4$ strings

1**

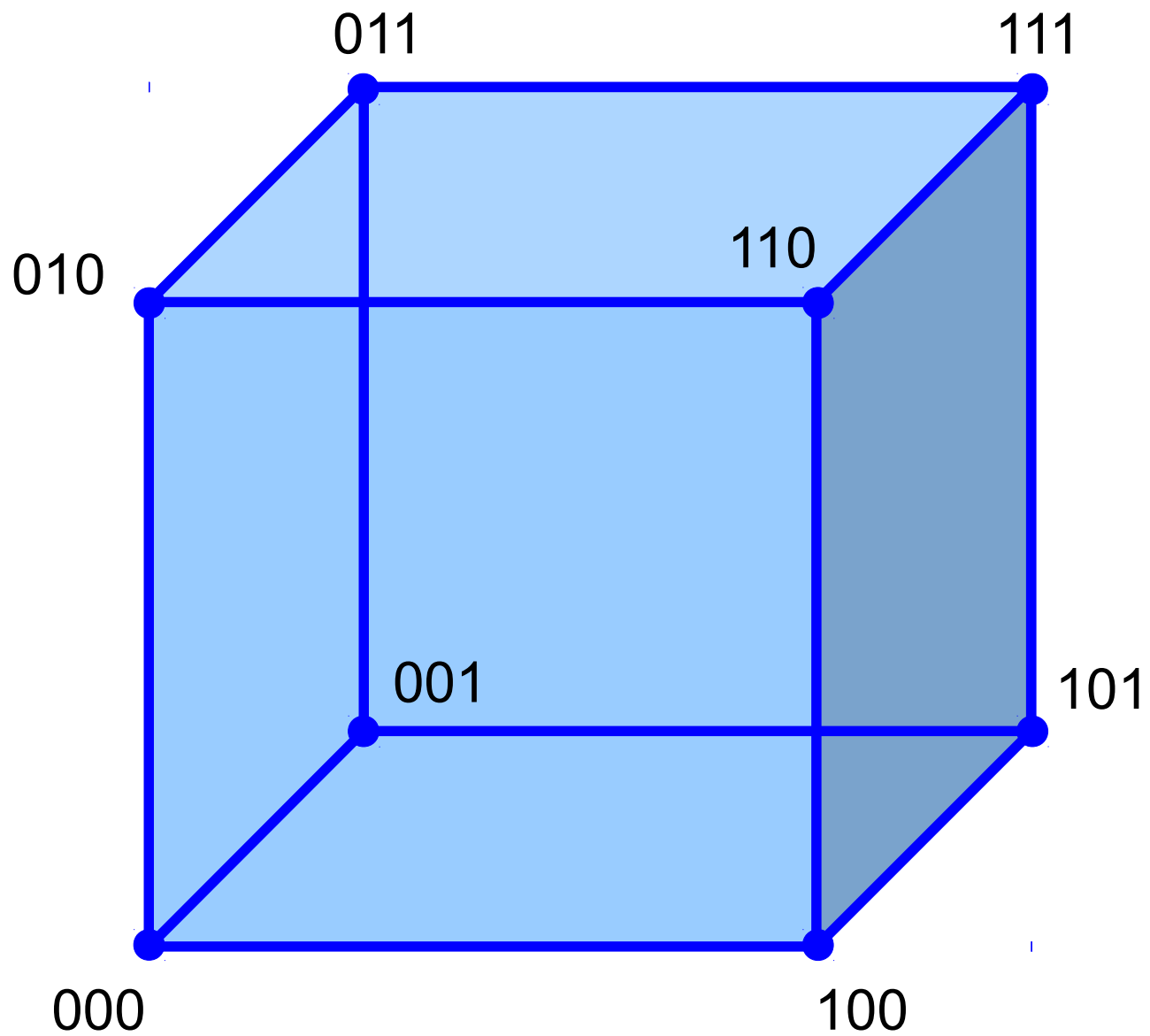


$2^2 = 4$ strings

*** 1 ***



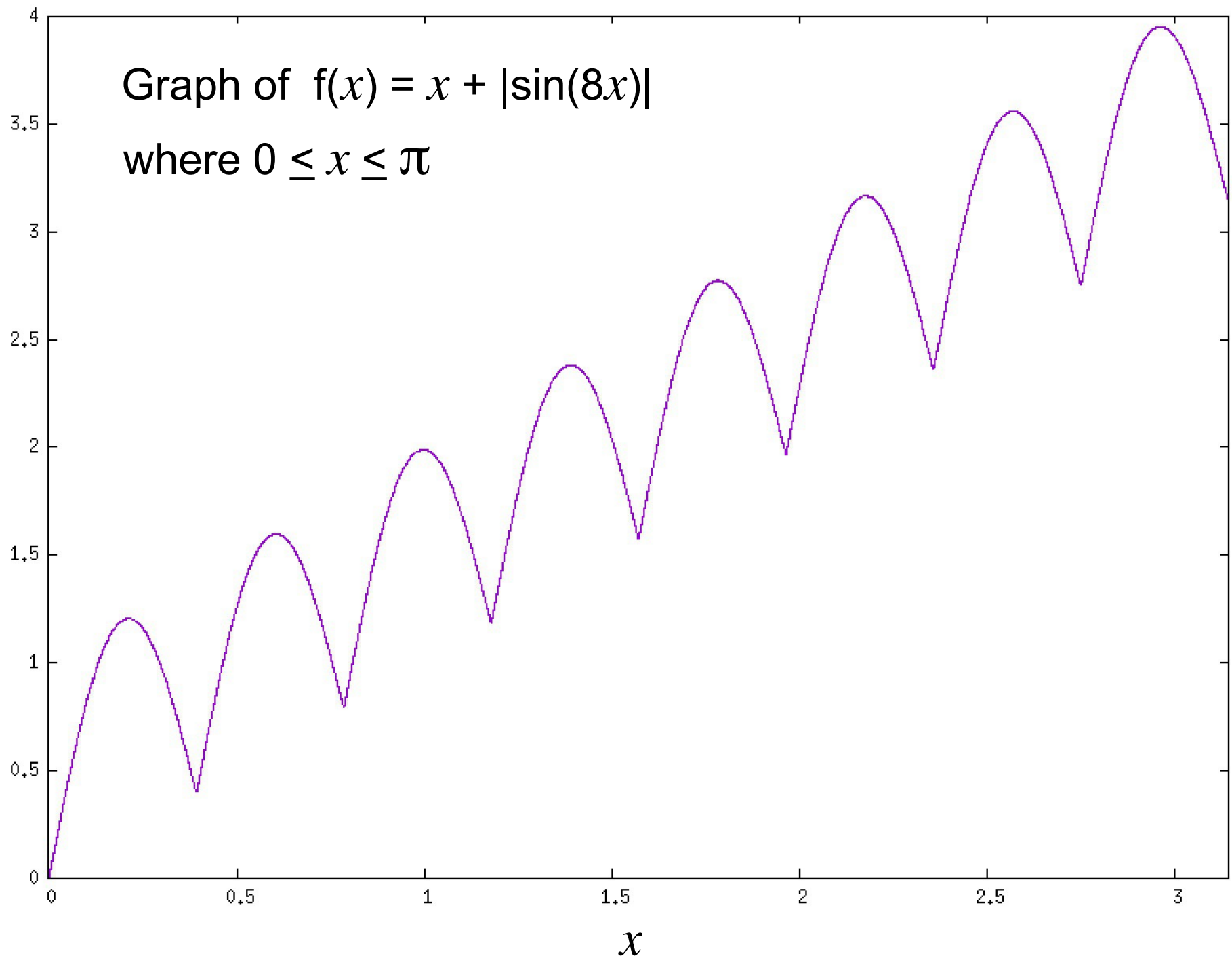
$2^2 = 4$ strings



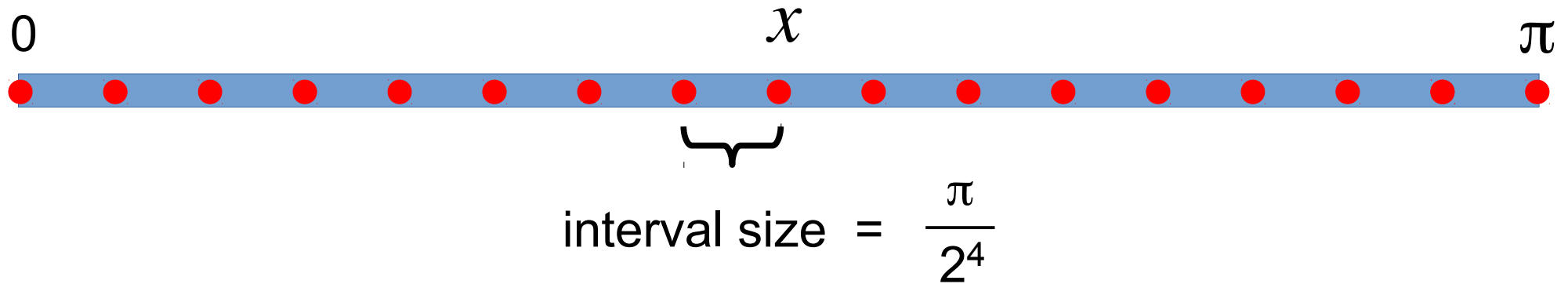
$2^3 = 8$ strings

Numerical Function Optimization with GAs

Graph of $f(x) = x + |\sin(8x)|$
where $0 \leq x \leq \pi$



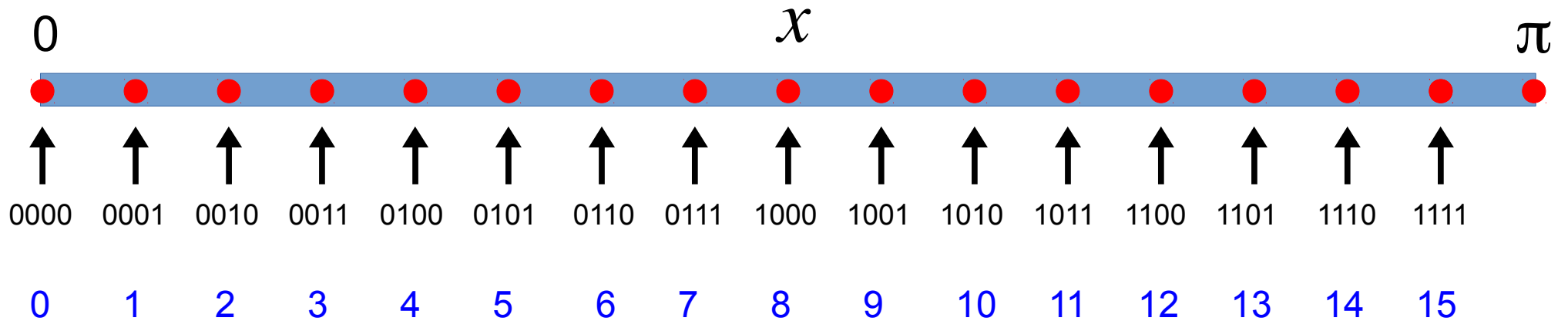
Representing Floating-Point Numbers



$2^4 = 16$ binary strings available

0000	0001	0010	0011	0100	0101	0110	0111
1000	1001	1010	1011	1100	1101	1110	1111

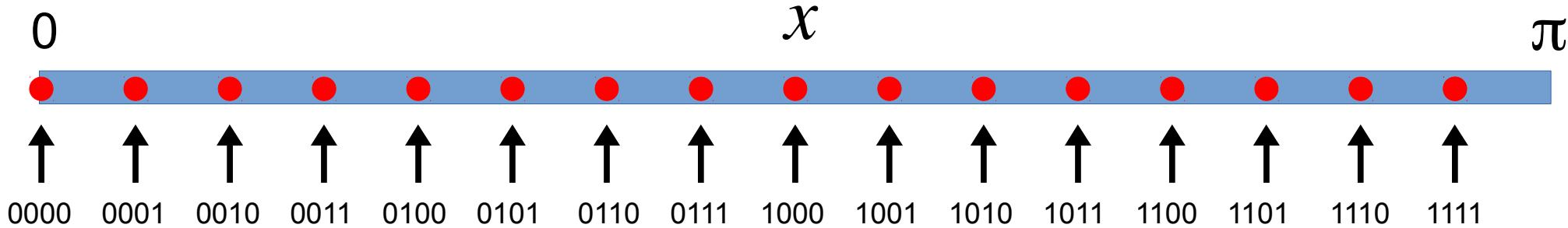
Representing Floating-Point Numbers



```
def genome_to_x(g):  
    x = binary_to_decimal(g) * math.pi / (2**len(g))  
    return x
```

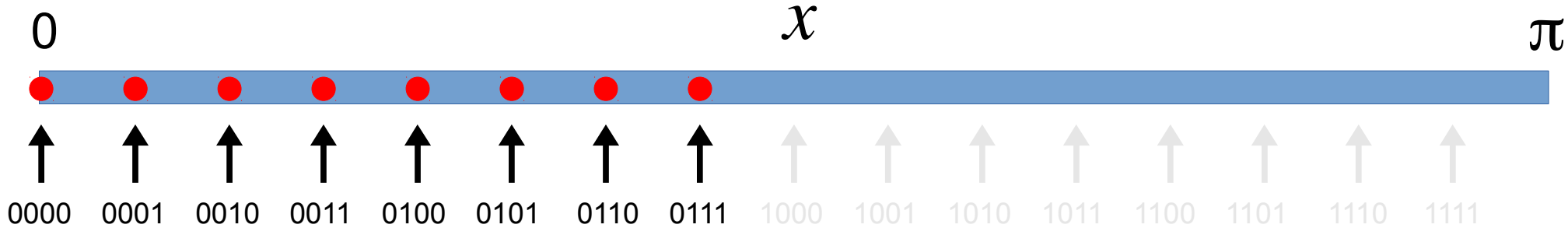
interval size

Schemas



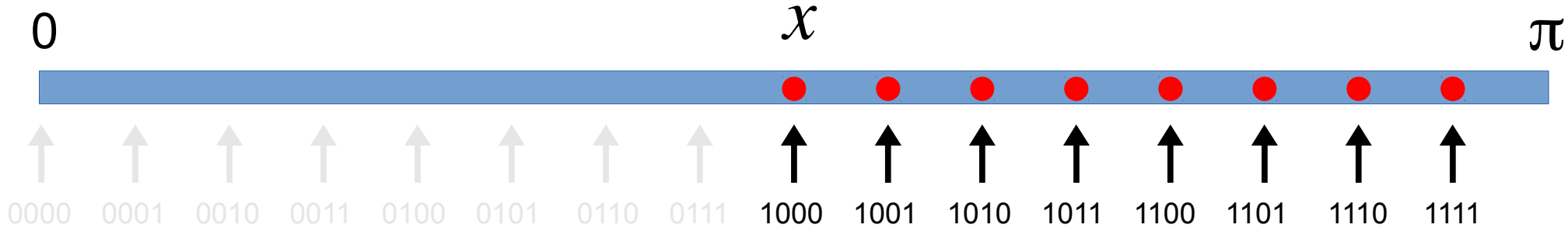
* * * *

Schemas



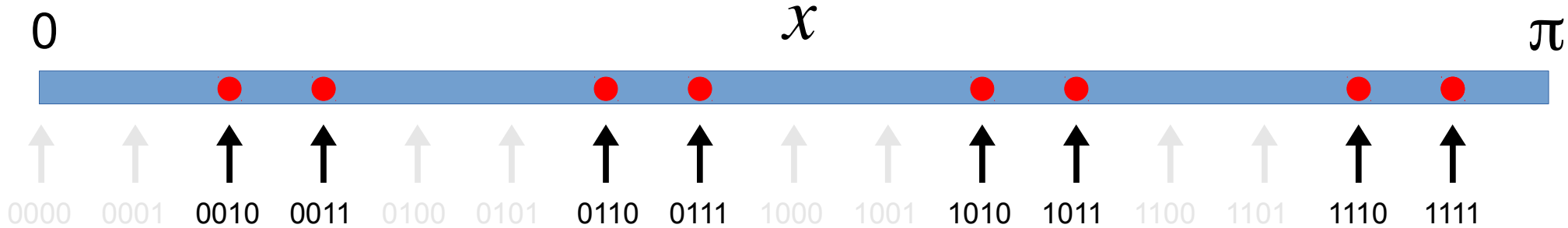
0***

Schemas



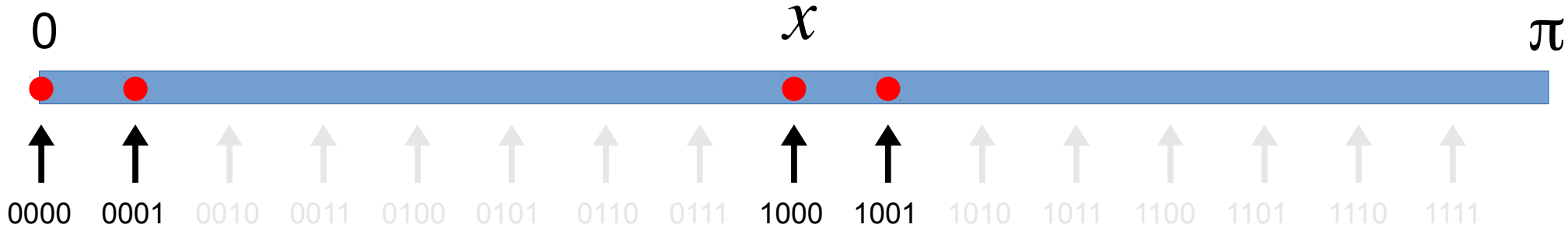
1***

Schemas



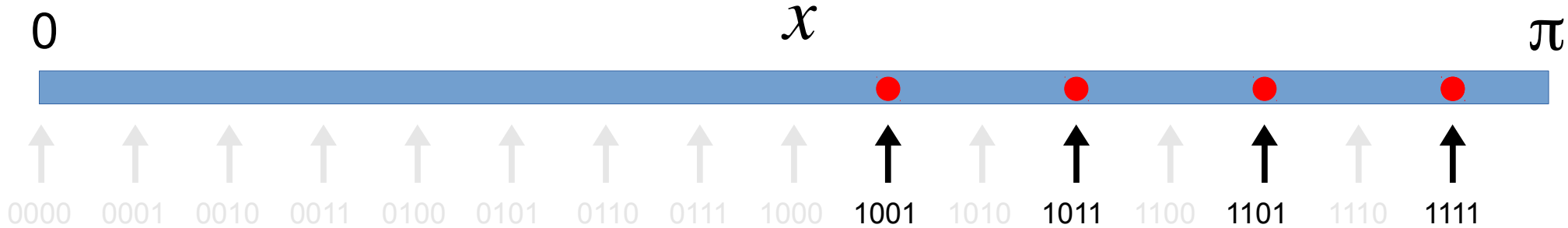
****1***

Schemas



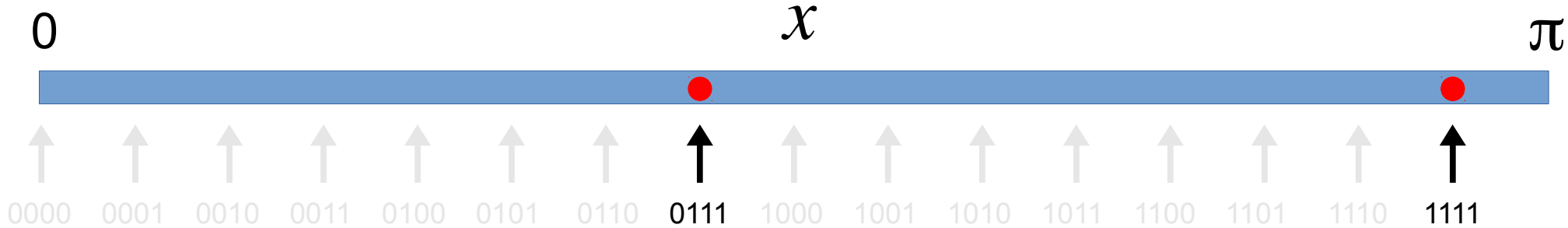
00

Schemas



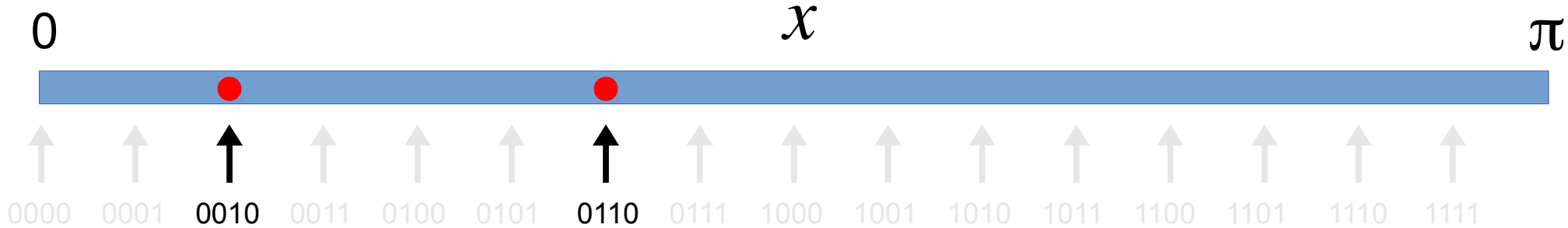
11**

Schemas



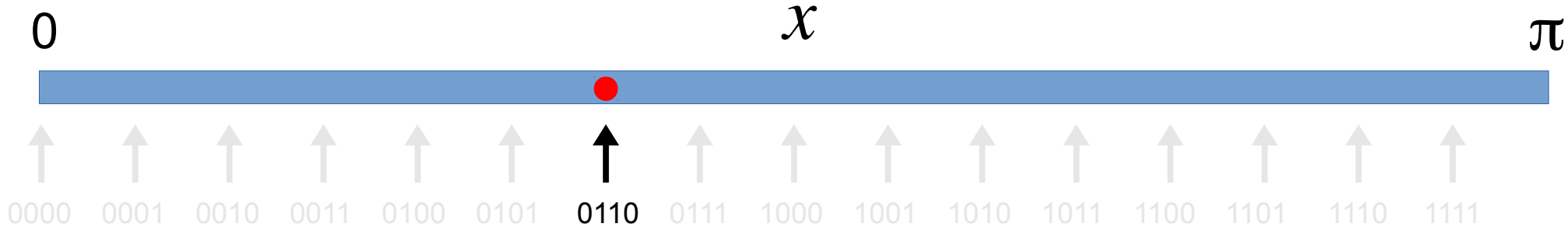
***111**

Schemas



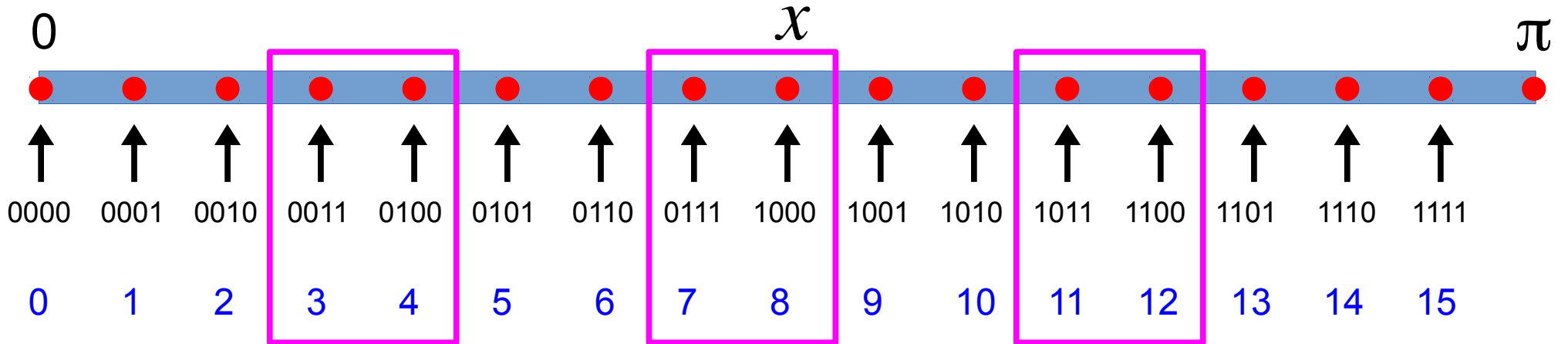
$0*10$

Schemas



0110

Problem: Representation Inconsistency



“number space”

7 ↔ 8

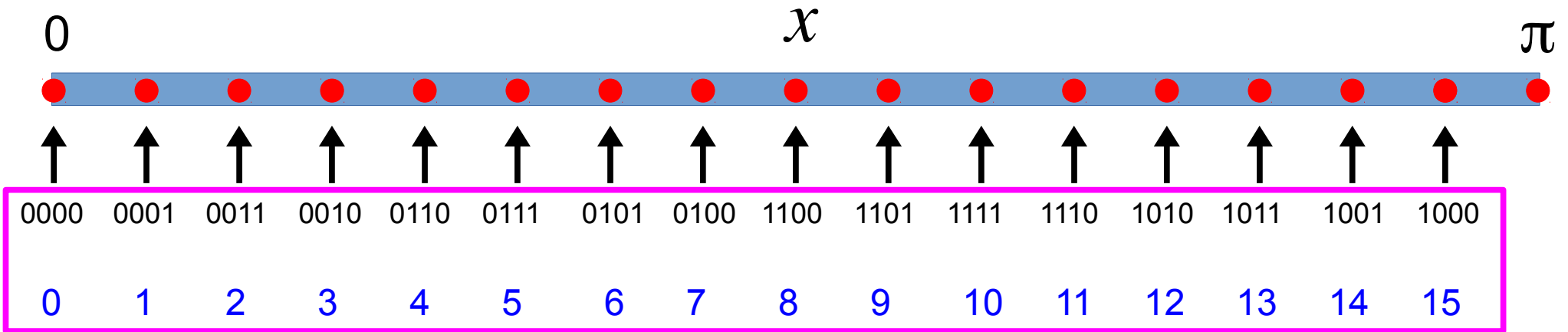
small jump

“genome space”

0111 ↔ 1000

big jump

Gray Code Representation of Numbers



“number space”

7 \longleftrightarrow 8

small jump

“genome space”

0100 \longleftrightarrow 1100

small jump