1)

A)

```
>> B = [1 0 0 0 0 ; 0.5 0 0.5 0 0; 0 0.5 0 0.5 0; 0 0 0.5 0 0.5; 0 0 0 0 1]'

B =

    1.00      0.50         0          0         0
       0         0      0.50          0         0
       0      0.50         0       0.50         0
       0         0      0.50          0         0
       0         0         0       0.50      1.00
```

B)

```
>> A = [0 1 0 0 0 ; 0.5 0 0.5 0 0; 0 0.5 0 0.5 0; 0 0 0.5 0 0.5; 0 0 0 1 0]'

A =

       0      0.50         0          0         0
    1.00         0      0.50          0         0
       0      0.50         0       0.50         0
       0         0      0.50          0      1.00
       0         0         0       0.50         0
```

C)

```
>> C = [0 1 0 0 0 ; 0.4 0 0.6 0 0; 0 0.4 0 0.6 0; 0 0 0.4 0 0.6; 0 0 0 1 0]'

C =

       0      0.40         0          0         0
    1.00         0      0.40          0         0
       0      0.60         0       0.40         0
       0         0      0.60          0      1.00
       0         0         0       0.60         0
```

D)

```
>> D = [1 0 0 0 0 ; 0.4 0 0.6 0 0; 0 0.4 0 0.6 0; 0 0 0.4 0 0.6; 0 0 0 0 1]'

D =

    1.00      0.40         0          0         0
       0         0      0.40          0         0
       0      0.60         0       0.40         0
```

0	0	0.60	0	0
0	0	0	0.60	1.00

2)

3rd column and 2nd row

A)

```
>> B^5

ans =

    1.0000    0.6875    0.3750    0.1875         0
         0         0    0.1250         0         0
         0    0.1250         0    0.1250         0
         0         0    0.1250         0         0
         0    0.1875    0.3750    0.6875    1.0000
```

Probability: 0.1250

B)

```
>> A^5

ans =

         0    0.3125         0    0.1875         0
    0.6250         0    0.5000         0    0.3750
         0    0.5000         0    0.5000         0
    0.3750         0    0.5000         0    0.6250
         0    0.1875         0    0.3125         0
```

Probability: 0.5

C)

```
>> C^5

ans =

         0    0.1869         0    0.0947         0
    0.4672         0    0.3290         0    0.2368
         0    0.4934         0    0.4474         0
    0.5328         0    0.6710         0    0.7632
         0    0.3197         0    0.4579         0
```

Probability: 0.3290

D)

```
>> D^5

ans =

    1.0000    0.5421    0.2368    0.0947         0
         0         0    0.0922         0         0
         0    0.1382         0    0.0922         0
         0         0    0.1382         0         0
         0    0.3197    0.5328    0.8131    1.0000
```

Probability: 0.0922

3)

A)
The behavior of the column vector shows there is equal probability of the chain ending on either side. Considering this is an absorbing matrix, the chain will likely end at one of the ends(position 1 or position 5)

B)
Likewise to the unbiased, absorbing transition matrix, there would be equal chance to end up in a position less than or greater than 3. This would likely be between position 2 and 4.

C)
The probability is split between either sides of 3, however, there is a higher probability of ending at a position to the right of position 3.

D)
Considering the bias and the absorbing characteristic of this matrix, there is a high probability the chain will end at position 5.

4)

```
>> x0 = [0;0;1;0;0]

x0 =

     0
     0
     1
     0
     0
```

A)

For very large odd and even exponential values, the vector consistently converges to:

xk =
0.5000
0
0.0000
0
0.5000

This supports my initial prediction that the system tends to end in either state 1 or state 5 with equal probability

B)

For large even exponents:

xk =
0.2500
0
0.5000
0
0.2500

For large odd exponents:

xk =
0
0.5000
0
0.5000
0

The initial prediction is only partially correct, as this behavior shows alternating distributions depending on the parity of k, favoring positions 2,3, and 4 as the final position.

C)

For large even exponents:

xk =
0.1231
0
0.4615

0
0.4154

For large odd exponents:

xk =
0
0.3077
0
0.6923
0

The introduced bias justifies the variability in the probabilities. Positions 3,4, and 5 are highly favored in this system's behavior, however, again, we have alternating distributions based on the parity of the exponent, which is likely due to reflecting boundaries

D)
For both large odd and even exponential values:

xk =
0.3077
0
0
0
0.6923

The process converges to a final distribution, indicating that positions 1 and 5 are absorbing, however position 5 is more likely to be the final position.

x1 = [0; 0.5; 0; 0.5; 0]

x1 =

0
0.5000
0
0.5000
0

A)

For very large odd and even exponential values, the vector consistently converges to:

xk =
0.5000

```
     0
0.0000
     0
0.5000
```

This supports my initial prediction that the system tends to end in either state 1 or state 5 with equal probability

B)

For large even exponents:

```
xk =
        0
   0.5000
        0
   0.5000
        0
```

For large odd exponents:

```
xk =
        0.2500
        0
   0.5000
        0
   0.2500
```

Despite the initial prediction accurately predicting the steady state vector after an even number of moves, this behavior shows alternating distributions depending on the parity of k, and no convergence to a steady state.

C)

For large even exponents:

```
xk =
        0
   0.3077
        0
   0.6923
        0
```

For large odd exponents:

```
xk =
    0.1231
```

```
     0
0.4615
     0
0.4154
```

My initial prediction only partially aligns with this, as the system favors transitions toward states 4 and 5, though the odd/even parity still introduces variation.

D)
For both large odd and even exponential values:

```
xk =
0.3538
     0
     0
     0
0.6462
```

The process converges to a final distribution, indicating that state 5 is absorbing and highly likely to be the final state. This supports my initial prediction.

Overall, the Markov chains with absorbing boundaries produced clearer outcomes. The distributions would consistently converge to the absorbed states, making the final position highly predictable. In contrast, the chains with reflecting boundaries displayed alternating distributions depending on exponential parity(odd or even). This oscillatory behavior made long-term predictions more variable and sensitive to initial conditions and the number of steps taken. Additionally, the choice of the starting vector has no influence on the final, steady state vector, as shown by the similar results, despite choosing two different starting vectors.

5)

A)

```
>> eigs(B)

ans =

  1.000000000000000
  1.000000000000000
  0.707106781186547
 -0.707106781186547
                  0

>> rref(B-eye(5))
```

ans =

```
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     0
     0     0     0     0     0
```

The matrix is diagonalizable. The eigenvalue, 1, with algebraic multiplicity 2 has two corresponding eigenvectors and the remaining eigenvalues are distinct, therefore the matrix is diagonalizable.

B)

```
>> eigs(A)

ans =

    1.0000
   -1.0000
   -0.7071
    0.7071
   -0.0000
```

The matrix is diagonalizable

C)

```
>> eigs(C)

ans =

   -1.0000
    1.0000
    0.6928
   -0.6928
   -0.0000
```

The matrix is diagonalizable

D)

```
>> eigs(D)

ans =

    1.0000
```

```
   1.0000
  -0.6928
   0.6928
        0

>> rref(D-eye(5))

ans =

     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     0
     0     0     0     0     0
```

The matrix is diagonalizable. The eigenvalue, 1, with algebraic multiplicity 2 has two corresponding eigenvectors and the remaining eigenvalues are distinct, therefore the matrix is diagonalizable.

- Transition matrices with absorbing boundaries tend to have repeated eigenvalues, making it more likely to be defective. On the other hand, the transition matrices with reflecting boundaries had distinct eigenvalues, therefore diagonalizable.
- The eigenvalues range between -1 and 1.
- The matrices are all non-invertible, because they all have an eigenvalue of 0