**Problem 1**

**Introduction**

**Methodology**

**Results**

|  |  |
| --- | --- |
| Probability Transmission Error |  |
| Ans. | p = 5906/10000 |

**Appendix**

import random

#globals

pknot = 0.4

epsilon = 0.03

epsilonknot = 0.02

# Determine Message

def message(m):

if( m <= pknot):

S = 0

else:

S = 1

return S

#Determine Recieved Message

def received(t,S):

if( S == 0 and t <= epsilonknot):

R = 1

elif( S == 0 and t > epsilonknot):

R = 0

elif( S == 1 and t <= epsilon):

R = 1

elif( S == 1 and t > epsilon):

R = 0

return R

# Main

failures = 0

count = 0

while(count < 10000):

count +=1

m = random.uniform(0,1) # generate a random number

S = message(m)

t = random.uniform(0,1)

R = received(t,S)

if( S != R):

failures +=1

print(failures)

**Problem 2**

**Introduction**

**Methodology**

**Results**

|  |  |
| --- | --- |
| Conditional Probability P(R = 1 | S = 1) |  |
| **P =** | **196** |

**Appendix**

#Alexander Fielding

# EE 381 Problem 2

import random

#globals

pknot = 0.4

epsilon = 0.03

epsilonknot = 0.02

# Determine Message

def message(m):

if( m <= pknot):

S = 0

else:

S = 1

return S

#Determine Recieved Message

def received(t,S):

if( S == 0 and t <= epsilonknot):

R = 1

elif( S == 0 and t > epsilonknot):

R = 0

elif( S == 1 and t <= epsilon):

R = 1

elif( S == 1 and t > epsilon):

R = 0

return R

# Main

success = 0

count = 0

while(count < 10000):

count +=1

m = random.uniform(0,1) # generate a random number

S = message(m)

if( S == 1):

t = random.uniform(0,1)

R = received(t,S)

if( S == R):

success +=1

print(success)

**Problem 3**

**Introduction**

**Methodology**

**Results**

|  |  |
| --- | --- |
| Conditional Probability P(S = 1 | R = 1) |  |
| **P =** | **188** |

**Appendix**

#Alexander Fielding

# EE 381 Problem 2

import random

#globals

pknot = 0.4

epsilon = 0.03

epsilonknot = 0.02

# Determine Message

def message(m):

if( m <= pknot):

S = 0

else:

S = 1

return S

#Determine Recieved Message

def received(t,S):

if( S == 0 and t <= epsilonknot):

R = 1

elif( S == 0 and t > epsilonknot):

R = 0

elif( S == 1 and t <= epsilon):

R = 1

elif( S == 1 and t > epsilon):

R = 0

return R

# Main

success = 0

count = 0

while(count < 10000):

count +=1

m = random.uniform(0,1) # generate a random number

S = message(m)

t = random.uniform(0,1)

R = received(t,S)

if( R == 1 and R == S):

success+=1

print(success)

**Problem 4**

**Introduction**

**Methodology**

**Results**

|  |  |
| --- | --- |
| Probability of error with enhanced transmission |  |
| **P =** |  |

**Appendix**