

In [10]:

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
```

IMPORTING DATA

In [8]:

```
df = pd.read_csv("Ads_Optimisation.csv")
df
```

Out[8]:

	Ad 1	Ad 2	Ad 3	Ad 4	Ad 5	Ad 6	Ad 7	Ad 8	Ad 9	Ad 10
0	1	0	0	0	1	0	0	0	1	0
1	0	0	0	0	0	0	0	0	1	0
2	0	0	0	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	1	0	0
4	0	0	0	0	0	0	0	0	0	0
...
9995	0	0	1	0	0	0	0	1	0	0
9996	0	0	0	0	0	0	0	0	0	0
9997	0	0	0	0	0	0	0	0	0	0
9998	1	0	0	0	0	0	0	1	0	0
9999	0	1	0	0	0	0	0	0	0	0

10000 rows × 10 columns

In [9]:

```
df.shape
```

Out[9]:

(10000, 10)

IMPLEMENTING THOMPSON SAMPLING

```
import random
N = df.shape[0]
d = df.shape[1]
ads_selected = []
number_of_rewards_1 = [0]*d
number_of_rewards_0 = [0]*d
total_reward = 0

for n in range(N):
    ad = 0
```

```

max_random_beta = 0
for i in range(d):
    random_beta = random.betavariate(number_of_rewards_1[i] +
1,number_of_rewards_0[i] + 1)
    if random_beta > max_random_beta:
        max_random_beta = random_beta
        ad = i
    ads_selected.append(ad)
    reward = df.iloc[n, ad]
    if reward == 1:
        number_of_rewards_1[ad] += 1
    else:
        number_of_rewards_0[ad] += 1
    total_reward += reward
print(total_reward)

```

In [37]:

```
print(ads_selected)
```

```

4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 0, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 9, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4]

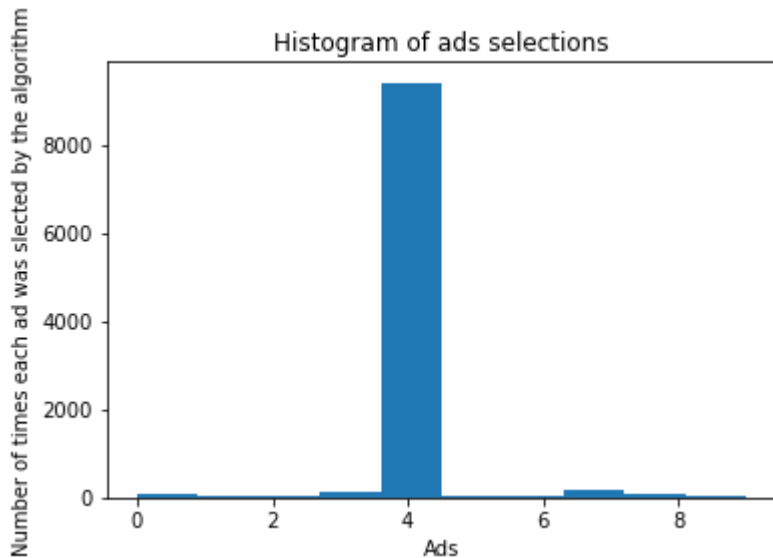
```

In [38]:

```
plt.hist(ads_selected)
plt.title('Histogram of ads selections')
plt.xlabel('Ads')
plt.ylabel('Number of times each ad was selected by the algorithm')
```

Out[38]:

```
Text(0, 0.5, 'Number of times each ad was selected by the algorithm')
```



REDUCING SIZE OF THE LIST

In [46]:

```
import random
N = df.shape[0]
d = df.shape[1]
ads_selected = []
number_of_rewards_1 = [0]*d
number_of_rewards_0 = [0]*d
total_reward = 0

for n in range(5000):
    ad = 0
    max_random_beta = 0
    for i in range(d):
        random_beta = random.betavariate(number_of_rewards_1[i] + 1, number_of_rewards_0[i] + 1)
        if random_beta > max_random_beta:
            max_random_beta = random_beta
            ad = i
    ads_selected.append(ad)
    reward = df.iloc[n, ad]
    if reward == 1:
        number_of_rewards_1[ad] += 1
    else:
        number_of_rewards_0[ad] += 1
    total_reward += reward
print(total_reward)
```

1277

In [44]:

```
print(ads_selected)
```

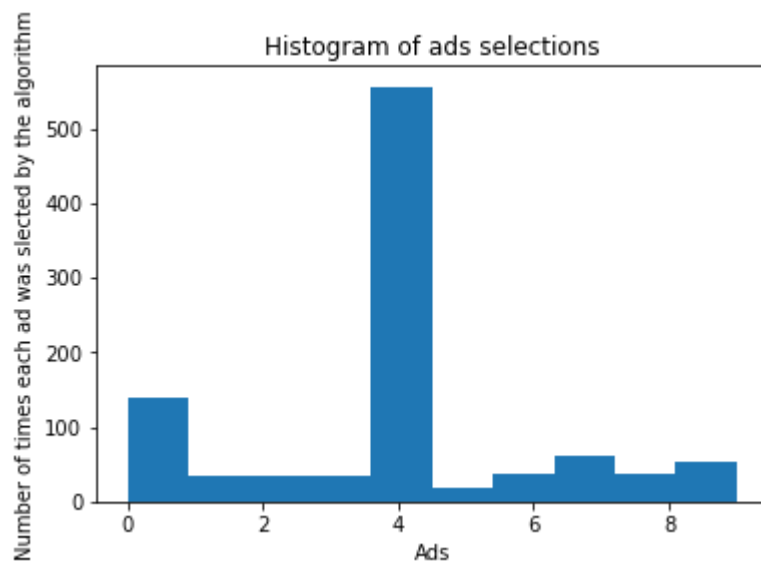
```
[4, 2, 1, 9, 5, 7, 5, 4, 3, 4, 3, 4, 2, 0, 6, 6, 8, 0, 1, 4, 4, 9, 3,
4, 3, 8, 8, 7, 1, 6, 9, 4, 9, 9, 4, 9, 9, 5, 1, 7, 7, 9, 7, 8, 9, 7,
6, 2, 4, 7, 5, 4, 0, 2, 7, 5, 9, 9, 4, 3, 1, 4, 7, 7, 3, 9, 8, 9, 7,
7, 4, 9, 8, 9, 7, 7, 8, 8, 5, 5, 3, 0, 4, 3, 4, 2, 0, 6, 1, 6, 8, 6,
6, 6, 2, 4, 2, 3, 5, 6, 3, 4, 2, 9, 9, 9, 7, 9, 2, 2, 9, 3, 2, 7, 3,
5, 0, 1, 3, 9, 4, 7, 2, 2, 1, 8, 9, 1, 7, 1, 9, 6, 7, 0, 4, 1, 6, 9,
6, 3, 4, 3, 6, 0, 2, 9, 6, 4, 8, 9, 7, 2, 3, 4, 6, 7, 7, 8, 1, 6, 9,
6, 5, 6, 6, 0, 4, 7, 7, 3, 6, 7, 2, 2, 9, 7, 9, 9, 3, 6, 6, 6, 4, 3,
9, 7, 4, 5, 7, 7, 6, 7, 2, 9, 6, 3, 2, 9, 7, 9, 1, 5, 2, 7, 4, 9, 0,
9, 9, 0, 8, 9, 6, 9, 1, 9, 3, 2, 7, 0, 3, 3, 7, 2, 7, 3, 4, 7, 0, 7,
0, 6, 2, 3, 2, 9, 4, 0, 0, 7, 9, 7, 4, 6, 0, 9, 7, 5, 7, 0, 0, 3, 7,
2, 9, 7, 3, 1, 2, 8, 8, 3, 6, 7, 1, 0, 4, 0, 3, 0, 9, 0, 4, 1, 2, 7,
5, 5, 9, 4, 4, 1, 1, 0, 9, 7, 2, 4, 2, 4, 1, 4, 3, 8, 2, 1, 0, 4, 4,
0, 4, 0, 7, 0, 9, 0, 8, 7, 0, 4, 4, 3, 0, 4, 0, 1, 0, 0, 1, 0, 0, 0,
0, 1, 0, 1, 0, 4, 0, 0, 0, 0, 0, 0, 0, 0, 9, 0, 0, 4, 0, 2, 0, 0,
7, 4, 4, 0, 4, 0, 0, 0, 9, 0, 5, 0, 0, 0, 0, 0, 4, 0, 4, 0, 0, 0, 2,
1, 0, 6, 7, 0, 7, 0, 4, 0, 0, 0, 0, 8, 0, 0, 7, 1, 0, 0, 0, 0, 0, 0,
1, 0, 0, 4, 0, 0, 4, 7, 0, 4, 4, 6, 0, 4, 4, 9, 4, 4, 8, 0, 4, 4, 0,
1, 9, 0, 0, 4, 4, 8, 4, 1, 6, 0, 5, 7, 0, 2, 9, 0, 4, 0, 4, 4, 0, 7,
0, 7, 6, 0, 4, 9, 8, 1, 0, 4, 0, 3, 3, 8, 0, 1, 4, 0, 0, 4, 4, 4, 0,
4, 4, 8, 3, 0, 0, 4, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 0, 4, 0, 4, 4,
4, 4, 4, 4, 4, 4, 4, 7, 4, 0, 4, 4, 8, 4, 4, 4, 4, 4, 4, 4, 4, 0,
8, 8, 0, 4, 4, 6, 4, 4, 4, 4, 0, 4, 0, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4,
4, 4, 1, 4, 4, 4, 4, 0, 0, 4, 2, 8, 4, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4,
8, 4, 4, 4, 4, 0, 5, 0, 4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 0, 4, 0, 4, 4, 4, 4, 4, 6, 4, 4, 4, 4, 4, 4, 4, 4, 0,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 8, 4,
4, 4, 7, 8, 4, 4, 4, 4, 4, 4, 1, 4, 4, 8, 4, 4, 4, 4, 4, 0, 4, 0, 7,
0, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 8, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 0, 8, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 4, 8, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 8, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 4, 8, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 7, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4]
```

In [45]:

```
plt.hist(ads_selected)
plt.title('Histogram of ads selections')
plt.xlabel('Ads')
plt.ylabel('Number of times each ad was selected by the algorithm')
```

Out[45]:

Text(0, 0.5, 'Number of times each ad was selected by the algorithm')



In []: