ST3009 Midterm

Samuel Petit - 17333946 - petits@tcd.ie

Table of Contents

Report

Please note: To do this and all of the other questions, I used matlab as well as the provided dataset. The code for all of the questions is available below this report.

Question 1 a

To answer this question, I started by selecting the requests time from the dataset for user 0. I then went through every single data point, setting the value in a newly created array of data of the same size a value of 0 or 1. 0 in the case that the request took less than 10ms and 1 otherwise.
After going through that data, I used the counts of 0s and 1s to plot the following graph using matlab's 'bar' function. To see the graph please see the appendix below this report.

Question 1 b

From question 1 a I already had an array of 0s and 1s and thus I only had to compute the mean of those values to get an estimation for Prob(X0 = 1).
We can do as such using matlab's built in 'mean' function or simply with : countOf1s / 1000 (as we have 1000 data points in this case). We find that our estimated value for P(X_0 = 1) is **0.2980**.

Question 1 c

We can apply CLT and estimate a 95% interval quite simply with the following:
1.96 * (stdDev / sqrt(size)) + meanUser0 <= *Prob(X0 = 1)* <=1.96 * (stdDev / sqrt(size)) + meanUser0.
Here, size is the amount of data points we have (1000), we can obtain the standard deviation from the variance by using matlab's function 'var' which returns the variance and take the square root value of that variance. (in matlab: sqrt(var(*<our 1s and 0s array>*))).
We can apply the Chebyshev inequality to build a 95% confidence interval in a similar manner with the following:
-stdDev / sqrt(0.05 * size) + meanUser0 <= Prob(X0 = 1) <= stdDev / sqrt(0.05 * size) + meanUser0
Finally, we can use the bootstrapping approach quite easily thanks to matlab's function 'bootci' which takes a size, set of parameters and our array of data (1s and 0s). This is done in my code with the following line:
ci = bootci(size, @mean, user0TransformedData);

We obtain the following confidence intervals:
Using CLT : 0.2696 <= P(X0 = 1) <= 0.3264
Using Chebyshev: 0.2333 <= P(X0 = 1) <= 0.3627
Using Bootstrapping: 0.2730 <= P(X0 = 1) <= 0.3290
Bootstrapping gives a full distribution and doesn't assume normality but it is an approximation when N is finite thus it can be uncertain as to how accurate it can be. It also

requires all measurements of N. In contrast, Chebyshev inequalities provide a bound and works for all N, however it is loose in general thus most likely not the most accurate. Finally, CLT gives a full distribution and only requires the mean and variance, however it is an approximation when N is finite the same way bootstrapping is.

Question 2

Using the same method as described in question 1 a and 1 b, we can apply this same algorithm when selecting specifically users 2 3 and 4s data and we obtain the following estimations:

$P(X_1 = 1) = 0.3440$

$P(X_2 = 1) = 0.2920$

$P(X_3 = 1) = 0.5470$

Question 3

Using the values provided as well as our computed estimates for $P(X_0 = 1)$, $P(X_1 = 1)$, $P(X_2 = 1)$, $P(X_3 = 1)$, as well as using marginalisation, we can obtain $P(Zn > 10)$.

$P(Zn > 10) = P(X_0 = 1) * P(Un = 0) + P(X_1 = 1) * P(Un = 1) + P(X_2 = 1) * P(Un = 2) + P(X_3 = 1) * P(Un = 3) = 0.3421$

Question 4

4. Calculate P(Un = 0|Zn > 10). Hint: Use Bayes Rule.

From Bayes rule we have:

- $P(R|F) = P(F|R)P(R) / P(F)$
- $P(F) = P(F \mid R)P(R) + P(F \mid R^c)P(R^c)$

Applying this to our specific probability:

- $P(Un = 0|Zn > 10) = P(Zn > 10 \mid Un = 0)P(Un = 0)/P(Zn > 10)$
- $P(Zn > 10) = P(Zn > 10 \mid Un = 0)P(Un = 0) + P(Zn > 10 \mid Un != 0)P(Un != 0)$

$P(Zn > 10 \mid Un = 0) = P(X0 = 1) = 0.2980$

$P(Zn > 10 \mid Un != 0) = 1 - P(Zn > 10 \mid Un = 0) = 1 - P(X0 = 1) = 1$ $- 0.2980 = 0.702$

$P(Un = 0)$ is provided in the data. $P(Un = 0) = 0.2710132564679$

$P(Un != 0) = 1 - P(Un = 0) = 1 - 0.2710132564679 = 0.72898674353$

Thus:

$P(Zn > 10) = P(X0 = 1) * P(Un = 0) + (1 - P(X0 = 1)) * (1 - P(Un = 0)) = 0.5925$.

Finally:

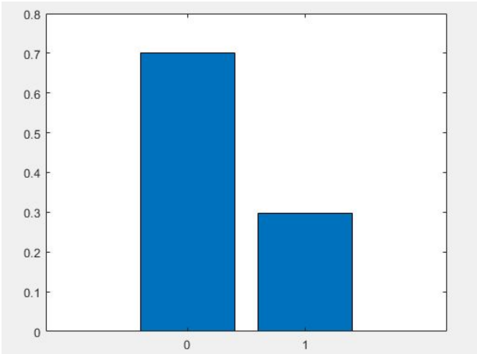
$P(Un = 0|Zn > 10) = P(X0 = 1) * P(Un = 0)/P(Zn > 10) =$ **0.1363**.

Question 5

To simulate the system, I go through a picked size of 20000. For every iteration I pick a user based on their provided probability and then based on my estimations for each users request time I simulate whether that specific request takes longer than 10ms or not. If it does, my count gets incremented and we can finally obtain an estimation for $P(Zn > 10)$ by simply doing : amountOfRequestsOver10ms / 20000.

Here is the result of one of my simulations: **0.3386**. By running this simulation multiple times, we find that the value is always quite close to our calculations from question 3 which gave us 0.3421. I have also observed that by incrementing the size of the simulation the returned value seems to get closer to 0.3421.

Appendix



Matlab code

```
% Provided in the first line of our data
userProbabilities = [0.2710132564679  0.34401907623591  0.26506075987615
0.11990690742004];

% Load data file (with the first line of probabilities removed for it to
% work in matlab).
data = load('data-noheader.txt');

% Select the first user's data
user0Data = data(:,1);

% Array with 1s and 0s to keep track of request timing over or under
10ms
user0TransformedData = (length(user0Data));

% Counts
more10msUser0 = 0;
less10msUser0 = 0;

size = length(user0Data);
for i=1:size
    % Add a 0 or 1 to the array based on the request time and increment
    % the appropriate count
    if(user0Data(i) > 10)
        user0TransformedData(i) = 1;
        more10msUser0 = more10msUser0 + 1;
    else
```

```
        user0TransformedData(i) = 0;
        less10msUser0 = less10msUser0 + 1;
    end
end
% Plot the graph
events = [0 1];
probabilities = [less10msUser0/size, more10msUser0/size];
bar(events, probabilities);

% Q1 B
meanUser0 = mean(user0TransformedData);
disp('the  estimation for P(X_0 = 1) is :');
disp(meanUser0);

% Q1 C
stdDev = sqrt(var(user0TransformedData));

% Computing the CI using  CLT
cltConfiDenceIntervalUser0 = [-1; 1] * 1.96 * (stdDev / sqrt(size)) +
meanUser0;
disp('CLT CI 95% : ');
disp(cltConfiDenceIntervalUser0);

% Computing the CI using  Chebyshev's Inequality
chebyConfiDenceIntervalUser0 = [-1; 1] * stdDev / sqrt(0.05 * size) +
meanUser0;
disp('chebyshevs CI 95% : ');
disp(chebyConfiDenceIntervalUser0);

% CI with bootstrapping has a provided function in Matlab
ci = bootci(size, @mean, user0TransformedData);
disp('Bootstrapping CI 95% : ');
disp(ci);


% Q2
% Select each user's data
user1Data = data(:,2);
user2Data = data(:,3);
user3Data = data(:,4);

% Array of 1s and 0s to transform our data
user1TransformedData = (length(user1Data));
user2TransformedData = (length(user2Data));
user3TransformedData = (length(user3Data));
```

```
% Count variables for each user
more10msUser1 = 0;
less10msUser1 = 0;

more10msUser2 = 0;
less10msUser2 = 0;

more10msUser3 = 0;
less10msUser3 = 0;

% Go through all the data
for i=1:size
    % For each user check if request takes more than 10ms
    % Set the obtained result in our transformed array and update counts
if
    % needed
    if(user1Data(i) > 10)
        user1TransformedData(i) = 1;
        more10msUser1 = more10msUser1 + 1;
    else
        user1TransformedData(i) = 0;
        less10msUser1 = less10msUser1 + 1;
    end

    if(user2Data(i) > 10)
        user2TransformedData(i) = 1;
        more10msUser2 = more10msUser2 + 1;
    else
        user2TransformedData(i) = 0;
        less10msUser2 = less10msUser2 + 1;
    end

    if(user3Data(i) > 10)
        user3TransformedData(i) = 1;
        more10msUser3 = more10msUser3 + 1;
    else
        user3TransformedData(i) = 0;
        less10msUser3 = less10msUser3 + 1;
    end
end

% Compute the mean (in this case it is our estimation) of our 1s and 0s
meanUser1 = mean(user1TransformedData);
meanUser2 = mean(user2TransformedData);
```

```
meanUser3 = mean(user3TransformedData);

disp('the  estimation for P(X_1 = 1) is :');
disp(meanUser1);

disp('the  estimation for P(X_2 = 1) is :');
disp(meanUser2);

disp('the  estimation for P(X_3 = 1) is :');
disp(meanUser3);

% Zn : vale is the time for a request
% Un index of the user submitted the nth request
% Q3
pZMore10ms = userProbabilities(1) * meanUser0 + meanUser1 *
userProbabilities(2) + meanUser2 * userProbabilities(3) + meanUser3 *
userProbabilities(4);
disp('The probability that Zn exceeds 10ms is:');
disp(pZMore10ms);

% Q4
% P(R|F) = P(F|R)P(R) / P(F)
% P(F) = P(F | R)P(R) + P(F | R^c)P(R^c)
% P(Un = 0|Zn > 10) = P(Zn > 10 | Un = 0)P(Un = 0)/P(Zn > 10)
% P(Zn > 10) = P(Zn > 10 | Un = 0)P(Un = 0) + P(Zn > 10 | Un != 0)P(Un
!= 0)

% P(Zn > 10 | Un = 0) = P(X0 = 1)
% P(Zn > 10 | Un != 0) = 1 - P(Zn > 10 | Un = 0) = 1 - P(X0 = 1)
% P(Un = 0) is provided in the data: userProbabilities(1)
% P(Un != 0) = 1 - P(Un = 0) = 1 - userProbabilities(1)
% Thus
% P(Zn > 10) =  P(X0 = 1) * P(Un = 0) +  (1 - P(X0 = 1)) * (1 - P(Un =
0))
% Finally :
% P(Un = 0|Zn > 10) = P(X0 = 1) * P(Un = 0)/P(Zn > 10)
pZnBayers = meanUser0 * userProbabilities(1) + (1 - meanUser0) * (1 -
userProbabilities(1));
pUn0GivenZnBayers = meanUser0 * userProbabilities(1) / pZnBayers;

disp('P(Un = 0|Zn > 10) is:');
disp(pUn0GivenZnBayers);


% Q5
```

```
% Building arrays where the 1st value is the probability of a request
being
% less than 10ms, the 2nd value being for a request taking more than
10ms.
probabilitiesUser0 = [(1 - meanUser0) meanUser0];
probabilitiesUser1 = [(1 - meanUser1) meanUser1];
probabilitiesUser2 = [(1 - meanUser2) meanUser2];
probabilitiesUser3 = [(1 - meanUser3) meanUser3];

countOver10ms = 0;
simulationSize = 20000;
for i=1:simulationSize
    % Pick a user based on provided probabilities, this returns a value
    % between 1 and 4 (the index for the user picked).
    randomlyPickedUser = randsample(4, 1, true, userProbabilities);
    isOver10ms = -1;
    switch randomlyPickedUser
        % Randomly (based on probabilities) pick a request to take less
        % than 10ms (return 0) or more than 10ms (return 1).
        case 1
            isOver10ms = randsample([0, 1], 1, true,
probabilitiesUser0);
        case 2
            isOver10ms = randsample([0, 1], 1, true,
probabilitiesUser1);
        case 3
            isOver10ms = randsample([0, 1], 1, true,
probabilitiesUser2);
        case 4
            isOver10ms = randsample([0, 1], 1, true,
probabilitiesUser3);
    end
    if(isOver10ms == 1)
        countOver10ms = countOver10ms + 1;
    end
end

disp('The stochastic simulation estimated a value for P(Zn > 10) of: ');
disp(countOver10ms/simulationSize);
```

Data Provided

#user probabilities: user 0: 0.2710132564679 user 1: 0.34401907623591 user 2: 0.26506075987615 user 3: 0.11990690742004

9 8 4 2
15 4 2 6
0 1 0 16
4 12 2 11
4 10 3 22
5 9 7 21
10 19 4 1
3 12 24 56
2 4 1 1
9 3 4 19
4 6 3 21
1 6 2 12
1 7 2 14
1 15 3 27
9 11 15 55
4 14 22 16
1 0 10 23
3 1 5 18
6 13 4 17
2 8 1 6
1 5 17 3
1 1 25 26
15 15 11 4
7 11 14 17
10 7 4 68
8 16 8 2
6 0 12 22
7 23 0 16
3 23 12 9
0 13 2 33
6 30 0 1
8 20 9 6
4 0 3 72
16 25 1 8
23 19 1 11
12 1 1 1
0 21 5 1
8 11 23 2
8 45 1 5
17 17 7 5
18 18 2 22

13 4 1 20
2 2 2 51
3 4 34 7
2 6 4 28
1 4 21 23
4 1 34 4
20 8 5 26
4 11 6 3
0 9 16 4
13 0 2 2
4 2 5 13
8 19 4 21
2 3 5 13
4 6 11 5
31 2 0 7
19 8 6 42
1 4 1 41
5 4 0 25
5 6 3 1
3 11 13 11
24 3 0 6
10 15 6 17
34 8 51 34
1 7 13 5
18 25 1 6
2 2 0 20
8 4 12 36
0 8 4 56
14 5 2 11
2 15 1 3
4 1 0 10
8 17 9 11
12 17 12 66
32 16 10 14
2 6 5 4
0 17 1 29
2 5 3 9
2 5 8 13
6 10 29 2
5 10 8 4
5 5 2 10
2 21 0 8
4 6 7 7
3 13 2 12
1 6 22 7
17 7 16 8

2 17 6 17
5 1 16 47
6 1 4 20
23 5 0 5
16 10 8 2
4 16 8 12
3 8 14 2
1 11 6 5
6 70 19 59
1 14 20 16
14 3 3 23
10 4 0 13
34 10 9 7
5 3 15 20
16 11 14 43
4 4 2 31
3 1 14 9
16 12 0 18
0 4 4 20
10 5 2 19
0 20 0 11
1 18 2 63
6 40 19 31
3 1 9 16
2 7 0 0
2 1 10 4
4 1 20 2
6 1 12 13
42 14 0 2
8 2 1 2
2 0 27 11
1 10 1 5
1 1 3 7
2 36 7 14
1 6 9 14
8 5 1 15
2 1 10 27
4 17 21 12
15 21 34 44
6 1 2 18
1 7 36 6
5 8 18 13
0 7 1 15
25 2 2 2
4 21 22 19
11 18 2 13

11 2 5 5
2 11 19 8
1 21 10 16
5 9 1 9
1 3 0 1
5 2 2 25
1 1 5 25
28 10 3 20
14 15 22 1
5 17 5 15
11 1 2 2
10 4 7 29
7 8 17 0
13 1 13 31
2 3 8 2
18 26 14 20
11 14 11 3
8 1 11 6
3 28 5 57
3 39 13 34
7 10 17 7
11 11 1 37
20 15 0 26
10 18 5 9
8 2 7 3
4 3 5 15
2 22 7 12
18 4 4 10
0 3 12 21
22 25 8 30
10 10 8 9
6 1 14 1
3 4 3 16
9 13 2 35
9 15 4 13
1 14 28 2
0 13 16 50
9 6 12 7
1 5 11 14
6 8 9 24
20 26 4 14
10 9 5 7
4 17 16 12
8 16 9 16
7 7 7 3
27 4 6 1

11 3 8 4
15 3 27 22
2 4 3 38
23 12 2 6
6 1 14 2
18 6 11 9
25 4 7 38
9 1 24 8
4 8 7 4
17 8 25 1
18 3 28 28
23 2 1 10
2 24 7 2
4 4 6 6
8 1 2 5
2 2 8 17
11 2 16 14
13 16 9 17
0 9 5 54
13 8 4 0
6 45 1 31
3 10 1 42
12 4 2 72
6 17 8 4
0 6 14 28
3 6 5 6
1 23 3 33
5 11 18 45
10 0 12 62
9 25 18 9
3 1 4 27
19 0 0 43
6 18 11 25
3 2 1 7
1 4 2 13
0 1 5 21
1 3 10 25
2 9 2 10
9 9 2 13
8 3 5 2
1 2 19 9
4 40 12 10
4 34 34 8
1 6 1 3
10 15 26 33
4 2 3 0

12 8 7 8
15 0 6 44
2 5 3 14
4 1 0 21
0 42 1 2
5 2 3 2
35 3 35 7
4 3 4 3
16 27 6 18
25 0 7 3
4 24 10 24
37 12 5 1
10 8 1 2
12 6 2 23
10 3 1 2
7 6 2 8
5 14 0 7
13 9 6 3
7 2 25 12
19 5 7 15
36 12 10 23
1 32 5 29
2 8 21 22
0 2 1 10
24 1 14 22
12 4 19 6
6 19 14 9
3 4 33 5
0 11 46 3
9 6 0 33
0 26 7 2
1 12 7 0
16 5 11 15
10 10 18 4
6 12 8 25
0 8 0 2
3 0 3 43
6 3 13 13
2 5 9 6
1 2 3 40
8 1 6 5
12 5 5 5
1 1 0 3
9 0 2 42
1 8 15 14
16 9 1 1

36 14 4 1
31 13 26 14
14 6 15 14
17 3 13 53
4 17 1 0
17 4 22 47
20 14 6 23
4 10 14 8
5 2 4 8
20 5 23 21
16 14 3 20
1 1 9 9
19 14 5 23
1 4 10 0
1 2 11 12
16 8 16 5
12 3 8 17
9 8 4 11
3 10 2 8
11 18 13 15
8 3 3 20
8 0 10 21
2 6 2 1
0 1 3 23
4 30 4 87
6 14 5 30
4 11 2 2
6 0 10 2
2 0 13 30
1 7 6 6
28 13 4 62
14 12 11 3
9 1 1 3
18 10 0 5
3 2 4 26
2 0 17 11
1 11 3 3
2 18 5 2
7 17 1 8
7 17 6 4
24 11 4 28
3 5 1 16
8 5 4 29
6 3 3 14
22 9 10 1
5 2 2 0

44 4 4 13
1 26 11 1
8 1 17 47
6 28 7 1
4 28 19 32
2 2 4 2
13 1 2 4
3 5 2 5
11 9 14 33
7 8 21 3
9 16 1 2
12 10 2 2
8 1 28 27
3 5 22 1
6 1 2 25
7 6 0 6
1 19 1 17
4 1 14 0
17 24 1 16
7 4 11 3
4 11 28 17
1 22 9 14
0 25 3 15
4 5 18 12
2 0 1 17
1 28 9 56
17 16 7 9
1 3 7 11
31 8 1 1
5 14 7 12
10 24 1 1
3 41 7 13
0 12 1 2
9 17 1 12
8 10 23 21
24 7 2 40
1 3 23 14
0 7 0 22
5 2 14 21
1 5 2 3
1 5 3 24
2 4 3 31
37 2 7 43
12 13 15 31
0 14 4 1
3 6 13 21

8 9 5 21
34 11 16 1
1 1 15 6
5 1 1 10
3 11 4 1
5 2 19 10
28 3 6 5
11 3 27 6
18 5 0 32
1 20 22 3
33 14 5 10
18 8 17 4
2 3 2 19
5 1 1 8
5 9 8 3
18 9 5 24
34 6 8 1
3 8 2 6
3 9 11 3
2 9 4 9
16 9 1 5
12 2 9 3
14 3 3 19
17 15 5 32
2 0 19 14
7 1 16 36
10 9 1 35
2 6 3 1
0 5 3 28
8 29 38 9
2 4 1 2
1 7 24 7
7 18 18 2
22 13 0 16
4 1 6 13
6 16 7 14
2 21 4 28
16 5 2 1
10 3 1 25
14 1 1 8
24 45 6 33
12 7 5 1
8 31 8 2
12 2 10 52
1 1 12 44
5 33 37 2

3 2 20 1
2 0 5 3
0 27 0 23
5 7 14 1
5 6 2 4
9 20 2 23
24 27 9 8
20 11 6 3
14 5 2 1
5 11 2 10
8 3 2 1
13 0 1 5
5 3 5 2
2 10 16 2
8 7 5 11
2 49 9 0
5 21 0 32
7 3 3 3
7 6 2 8
6 4 8 43
9 1 0 27
12 20 19 7
4 5 14 15
4 6 8 26
4 10 9 37
26 26 0 13
4 3 18 36
8 7 13 5
8 17 0 7
10 30 10 2
4 7 10 21
29 3 6 6
25 1 2 37
1 3 4 11
6 3 4 2
14 2 4 9
33 5 12 19
4 6 12 21
24 6 4 39
12 24 2 20
0 5 7 2
9 0 8 3
3 31 3 2
1 10 6 2
0 3 12 9
10 8 1 17

0 7 6 24
5 13 5 12
1 0 8 26
0 2 22 7
2 2 5 6
15 31 4 33
2 2 3 44
10 5 7 20
22 1 5 8
16 25 17 86
28 7 12 0
11 12 4 38
19 15 2 21
12 8 7 61
11 18 19 9
2 6 1 3
7 9 6 9
6 5 4 11
18 2 5 15
19 17 5 26
8 11 6 7
2 1 3 58
19 3 4 2
10 7 3 5
1 16 9 17
2 1 1 1
3 8 8 3
4 1 25 80
13 6 4 0
6 11 29 8
20 4 6 15
19 15 16 0
8 7 9 25
8 13 15 38
3 6 1 20
7 0 10 7
5 0 3 53
11 4 4 16
10 19 2 14
9 1 7 5
8 2 0 4
27 18 1 5
3 2 27 35
2 10 2 18
10 8 8 8
16 30 44 13

14 2 1 10
3 11 10 31
19 12 0 31
7 2 1 33
3 1 11 12
11 28 1 8
6 14 13 7
11 15 17 12
30 30 31 4
8 11 1 12
4 1 3 45
2 6 13 14
6 7 0 4
5 2 8 52
29 5 3 19
2 2 2 2
1 2 4 22
16 5 2 4
6 8 1 25
0 28 3 12
5 5 11 44
8 4 18 14
12 2 2 42
3 5 0 10
8 5 1 9
15 5 9 7
25 15 1 51
11 6 5 4
15 2 1 9
7 61 32 6
2 2 2 12
8 3 22 4
10 1 8 12
5 3 17 9
1 51 4 27
5 18 38 5
0 1 10 16
1 11 21 5
18 2 12 12
5 12 10 2
13 4 9 5
8 7 9 19
6 0 5 52
17 6 2 41
7 25 6 19
8 6 25 11

8 12 29 1
5 11 1 4
0 16 5 16
3 1 3 30
2 13 13 1
2 0 24 24
24 6 5 13
20 5 51 12
1 33 6 13
9 11 11 18
6 2 3 27
3 8 8 12
3 5 6 11
5 6 2 8
23 12 20 30
1 23 3 3
18 19 10 12
7 6 1 0
9 5 15 54
31 6 5 5
20 38 12 7
5 0 11 8
11 10 2 26
7 8 28 9
4 11 22 47
1 1 28 22
6 15 10 12
3 2 12 3
1 7 6 14
6 2 25 4
17 2 0 8
10 40 14 2
9 13 0 24
16 0 6 8
14 2 16 1
4 9 2 11
1 12 0 18
23 47 18 26
1 22 2 32
18 41 25 13
11 43 3 14
39 15 21 26
35 15 15 7
4 0 13 12
12 17 1 19
15 0 5 58

24 10 14 29
8 14 4 7
12 10 20 23
5 17 6 9
8 4 11 80
5 7 6 3
4 11 16 3
10 4 1 17
25 31 5 14
11 16 20 11
5 17 2 39
9 14 3 5
1 1 2 13
8 13 10 54
4 8 1 2
18 3 11 29
2 2 4 53
43 4 5 10
1 11 4 25
5 11 0 15
12 3 1 8
0 19 3 9
5 6 6 6
11 3 1 50
5 7 20 10
15 4 18 47
0 3 10 11
23 13 1 20
0 2 1 1
0 5 5 11
16 45 12 14
2 22 6 20
4 5 1 4
12 0 1 27
3 20 3 4
8 4 19 17
7 31 9 16
4 1 1 3
1 19 11 8
12 3 1 3
9 2 4 9
2 8 2 11
5 7 10 0
17 9 9 9
7 4 0 37
5 1 1 14

40 16 24 23
0 0 19 20
2 3 1 9
15 4 21 4
15 9 2 19
2 23 1 24
2 1 9 19
3 12 2 5
6 2 39 13
2 22 2 9
2 4 15 68
10 14 2 11
4 6 2 15
4 3 3 15
4 37 2 18
10 7 19 4
10 21 11 40
13 40 3 49
4 2 22 0
21 1 6 6
8 11 17 48
23 2 6 6
2 15 28 35
3 19 1 65
12 16 18 13
13 2 5 2
4 5 1 5
7 10 7 2
3 6 0 4
9 8 6 30
4 7 10 17
4 14 8 2
7 7 9 8
10 1 6 1
6 9 3 1
2 21 1 13
3 2 12 59
13 3 14 3
14 7 12 3
3 5 3 32
18 26 2 14
0 12 8 13
3 28 7 13
15 13 1 3
26 18 1 22
1 25 17 51

5 10 1 21
0 5 7 40
4 14 5 9
5 1 24 25
0 7 3 33
3 5 0 4
3 1 25 9
1 17 4 11
6 6 2 12
20 14 4 5
7 21 9 93
34 9 12 0
1 1 17 8
6 22 8 25
0 8 2 2
4 7 7 34
51 15 4 38
7 15 1 1
10 16 1 18
4 18 9 12
24 8 20 10
5 7 2 11
1 5 7 11
20 1 4 38
21 13 13 10
6 28 6 3
12 9 15 2
5 7 9 8
0 8 14 11
1 0 26 3
4 7 8 6
9 37 2 14
10 49 0 4
24 7 4 19
1 3 14 13
12 6 20 27
0 12 3 1
12 3 2 3
12 20 5 8
16 10 23 13
9 27 12 14
5 2 2 15
4 1 1 2
6 0 20 12
12 2 8 11
5 18 8 3

10 1 7 13
11 7 1 7
5 18 22 38
35 2 5 7
2 9 5 23
8 5 3 7
7 20 15 129
10 6 2 9
6 13 24 5
2 0 7 18
18 7 7 32
10 0 2 47
8 11 10 4
1 31 7 17
15 5 8 12
18 16 21 9
7 20 8 24
20 2 4 25
12 32 8 10
34 17 4 15
6 1 14 15
0 5 2 32
15 18 4 21
9 3 1 4
2 13 25 1
27 4 15 20
3 6 1 6
3 7 19 26
8 13 4 17
1 8 5 59
5 18 9 1
1 15 2 6
6 2 4 11
7 2 2 39
9 4 2 40
16 1 10 10
16 1 1 62
9 8 24 1
4 10 1 13
4 5 15 42
8 2 4 3
4 8 0 0
18 3 21 22
4 9 1 4
10 2 2 6
12 1 4 3

12 7 9 3
18 7 1 10
11 6 8 1
0 11 1 41
23 0 8 6
9 16 6 6
0 12 8 26
15 0 22 15
6 9 11 8
1 16 14 29
3 6 19 6
1 0 2 1
0 19 4 18
9 19 31 19
7 9 5 8
8 5 32 2
37 12 5 2
6 3 21 28
11 17 1 29
15 4 16 32
2 2 6 32
1 7 6 17
1 36 0 21
4 0 14 7
11 3 5 5
7 4 1 6
2 2 1 0
6 37 17 21
2 5 7 5
17 1 18 51
0 24 9 9
21 6 12 17
12 2 19 6
7 30 8 23
1 12 12 17
10 9 4 34
38 23 1 31
35 0 13 1
21 6 5 30
21 2 5 18
4 4 27 48
3 8 9 10
3 6 1 51
0 5 14 1
3 28 18 6
25 11 4 33

1 16 10 1
43 2 10 8
6 10 3 31
2 0 3 11
5 4 5 11
13 3 6 2
2 5 5 2
1 15 33 4
7 11 3 14
15 31 24 7
8 3 2 29
3 6 3 20
15 12 2 16
27 12 11 3
1 2 3 4
23 3 5 12
21 12 1 23
9 0 1 5
3 3 1 13
12 5 2 15
1 6 2 1
1 25 2 5
1 6 6 118
10 25 6 15
8 8 3 5
7 6 6 30
11 9 4 11
31 9 0 1
0 3 1 3
1 3 5 4
11 27 2 9
17 13 0 11
3 4 11 32
13 3 9 12
16 0 28 29
9 0 18 20
2 37 17 9
3 4 8 62
3 20 4 2
8 5 7 35
11 4 11 12
7 5 0 9
5 39 2 2
39 1 13 5
1 11 8 7
41 3 3 5

1 12 4 22
1 34 8 30
3 4 4 33
11 4 3 3
4 7 3 7
9 1 7 22
11 2 1 21
6 5 27 16
1 4 4 4
3 31 0 15
3 5 5 0
12 13 2 2
2 6 5 37
6 45 8 4
2 12 20 19
1 20 3 5
2 10 5 12
8 7 1 38
18 8 19 16
3 1 10 13
6 1 5 66
1 1 2 6
12 1 71 36
26 4 1 3
28 8 10 14
0 16 5 45
4 4 7 19
5 3 2 44
4 10 20 12
10 1 15 10
2 17 3 2
1 6 3 2
0 35 9 1
13 1 0 3
6 8 14 44
15 0 16 4
10 14 11 8
16 8 15 0
4 2 1 12
9 6 8 18
4 3 11 3
4 14 3 39
3 1 15 2
2 11 3 32
5 1 2 6
8 1 3 73

2 6 6 35
19 1 6 6
4 3 5 25
0 16 9 7
24 7 6 2
2 17 25 16
19 2 21 11
2 3 5 11
10 38 4 15
1 22 18 10
2 4 2 4
1 22 7 2
4 1 2 14
18 2 2 25
3 1 2 14
4 13 0 1
11 5 8 27
3 7 2 11
16 6 10 0
4 3 1 25
3 3 3 4
5 8 26 25
9 2 4 74
26 5 0 19
14 10 5 2
1 12 7 59
2 3 29 14
6 1 3 29
4 11 0 28
2 32 7 33
2 26 16 2
3 20 13 2
6 3 2 16
31 16 7 3
1 8 11 14
9 1 4 75
12 5 13 59
4 5 15 13
3 11 9 17
7 6 12 1
10 0 9 19
14 3 15 34
15 8 5 10
9 17 5 9
2 3 4 14
8 2 1 45

19 14 7 10
2 4 5 38
4 2 7 3
5 5 0 5
32 6 10 17
3 1 0 12
4 5 1 70
8 2 21 12
22 7 23 2
16 5 0 4
7 8 4 75
31 5 2 50
16 24 7 2
0 9 8 9
0 11 4 17
16 4 1 25
14 1 18 18
6 22 15 1
5 2 0 10
8 3 4 9
2 31 6 5
7 2 9 14
16 0 2 20
4 4 5 3
4 4 14 37
2 10 25 21
16 0 1 10
3 7 16 11
6 5 10 1
7 4 8 8
5 17 1 25
1 8 22 14
19 13 5 2
4 5 16 4
4 45 9 20
6 7 2 0
22 21 0 11
2 11 17 18
6 28 5 18