

|  |  |
| --- | --- |
| name | id |
| Mohamed Hossam El-Din El-Sayed | 1808581 |
| Mohamed Hesham Alsaied Albendary | 1808442 |
| Ahmed Salah Abdelhak | 1803820 |
| Ahmed Reda Ibrahim Taha Elawam | 1802745 |
| Mohamed Ashraf Farouk Ibrahim | 1805188 |
| Mohamed Ahmed Mahmoud Asran | 1808040 |

Ain Shams University  
Faculty of Engineering  
Computer and Systems Engineering  
CSE332s: Design and Analysis of Algorithm

Contents

[Task 1 1](#_Toc103623337)

[Assumptions 1](#_Toc103623338)

[Problem Description 1](#_Toc103623339)

[Detailed Solution 2](#_Toc103623340)

[Complexity Analysis 5](#_Toc103623341)

[Comparison With Another Algorithm//TODO: INSERT THE REFERENCE 6](#_Toc103623342)

[Sample Output 6](#_Toc103623343)

[Conclusion 7](#_Toc103623344)

[Task 4 9](#_Toc103623345)

[Assumptions 9](#_Toc103623346)

[Problem Description 9](#_Toc103623347)

[Detailed Solution 10](#_Toc103623348)

[Complexity Analysis 12](#_Toc103623349)

[Comparison With Another Algorithm 12](#_Toc103623350)

[Sample Output 13](#_Toc103623351)

[Conclusion 14](#_Toc103623352)

[Task 6 16](#_Toc103623353)

[Assumptions 16](#_Toc103623354)

[Problem Description 16](#_Toc103623355)

[Detailed Solution 16](#_Toc103623356)

[Complexity Analysis 20](#_Toc103623357)

[Comparison With Another Algorithm 20](#_Toc103623358)

[Sample Output 21](#_Toc103623359)

[Conclusion 24](#_Toc103623360)

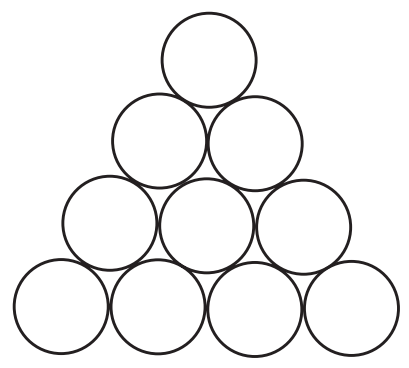
//TODO: INSERT TABLE OF FIGURES, TABLES

# Task 1

## Assumptions

User will input correct data type whenever they are asked for input (ex. When asked to enter number of rows, they won’t enter a string for example). Also, the centers of the coins are assumed to be at the points of an equilateral triangular lattice

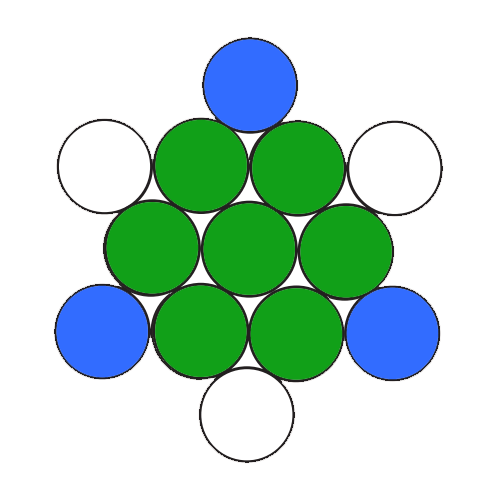
## Problem Description

Inverting a Coin Triangle Consider an equilateral triangle formed by closely packed pennies or other identical coins like the one shown in Figure 1. It’s required to use iterative improvement method to design an algorithm to flip the triangle upside down in the minimum number of moves if on each move it’s possible to slide one coin at a time to its new position.

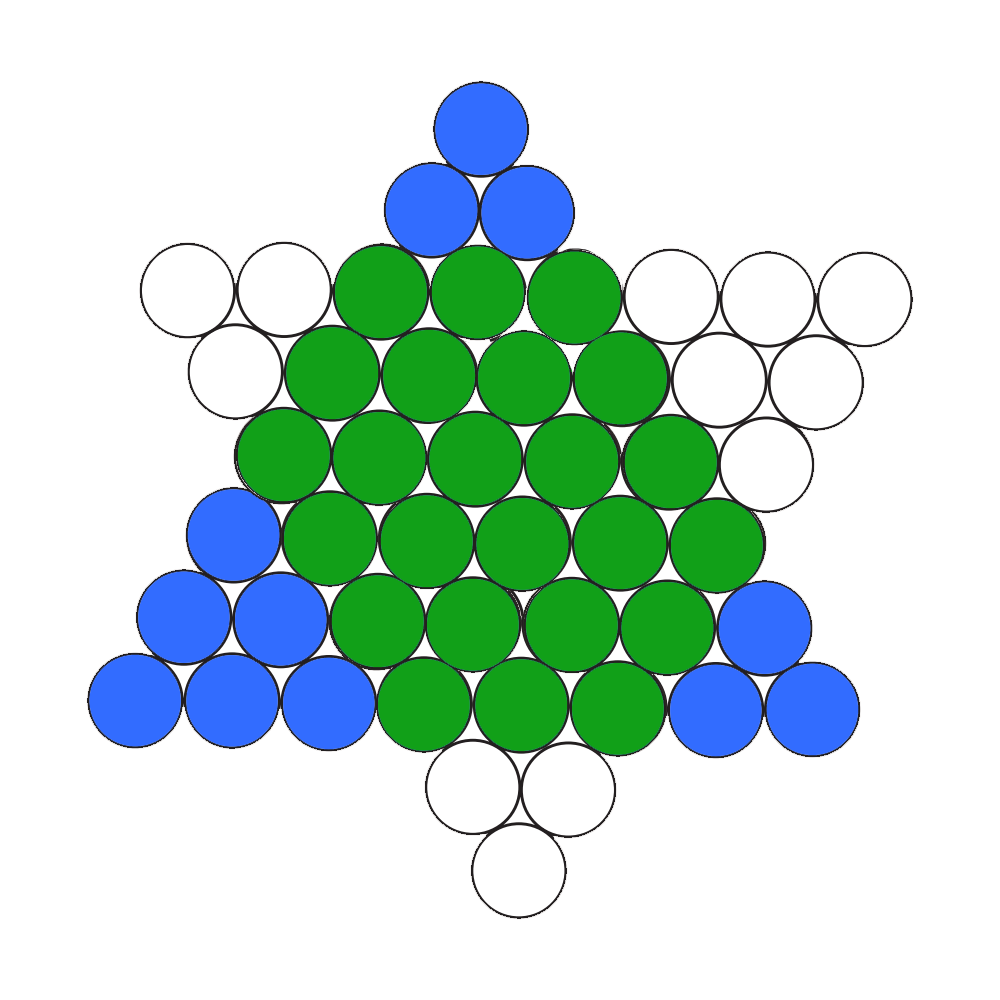
Figure

## Detailed Solution

A trivial solution to flip a triangle with rows is to just move all the coins to their new positions, which takes number of steps equal to number of coins in that triangle or 10 steps when applied to Figure 1(obviously inefficient!)

To flip the pyramid in the least number of moves, we will try to overlap both starting and ending positions together, this way the coins marked in green in Figure 2,will not be moved at all. And the ones marked in blue only will be moved to their new positions “Marked with white”. So, in Figure 1, we will need to make only 3 moves

Figure

Let’s draw this triangle with 8 rows

Figure

If we keep on drawing more and more triangles, we will find out that

1. The small blue triangle in the upper corner can always be moved to the corresponding small white triangle in the bottom
2. The small blue triangle in the bottom left corner can always be moved to the corresponding small white triangle in the upper right corner
3. The small blue triangle in the bottom right corner can always be moved to the corresponding small white triangle in the upper left corner

Table 1 shows the relation between number of rows and the number of rows of each of the small triangles

Table

|  |  |  |  |
| --- | --- | --- | --- |
| Number of rows | Number of rows in left triangle | Number of rows in upper triangle | Number of rows in right triangle |
| 1 | 0 | 0 | 0 |
| 2 | 1 | 0 | 0 |
| 3 | 1 | 1 | 0 |
| 4 | 1 | 1 | 1 |
| 5 | 2 | 1 | 1 |
| 6 | 2 | 2 | 1 |
| 7 | 2 | 2 | 2 |
| 8 | 3 | 2 | 2 |

It’s easy to prove that using data obtained in Table 1

Pseudo-code

ALGORITHM InvertTriangle(n)

    //INPUT: n - number of rows

    //OUTPUT: Inverted triangle

    //REQUIREMENTS: n > 0

    a <- ⌊(n + 1) / 3⌋

    for i <- 1 to do

        move one coin from bottom left triangle to upper right

        print triangle

    b <- ⌊(n) / 3⌋

    for i <- 1 to do

        move one coin from top triangle to bottom

        print triangle

    c <- ⌊(n - 1) / 3⌋

    for i <- 1 to do

        move one coin from bottom right triangle to upper left

        print triangle

    steps <- a + b + c

    return steps

## Complexity Analysis

Complexity for each function

printTriangle:

init:

sum:

options:

mainmenu:

To calculate total complexity, recall that we make moves and in each move, we print the triangle. So, in total complexity is , since are all functions in , total complexity is . However, the problem can be solved in if we calculate only number of steps as without printing the triangle after each step

## Comparison With Another Algorithm//TODO: INSERT THE REFERENCE

Another way to calculate number of steps which I found online is by using this formula , this formula and my solution give the same number of steps for all values of

## Sample Output

Figure

Now we will try but since it will produce a very long output, I’ll go to options and disable step by step output so that we can see only number of steps as shown in Figure 5

## Conclusion

Figure

Inverting a triangle of coins using this algorithm uses iterative improvement since on each step we are getting closer to the goal result by doing the same step over and over (moving one coin between two small triangles)

# Task 4

## Assumptions

The user will input a positive integer number of pennies.

## Problem Description

A “machine” consists of a row of boxes. To start, one places n pennies in the leftmost box. The machine then redistributes the pennies as follows. On each iteration, it replaces a pair of pennies in one box with a single penny in the next box to the right. The iterations stop when there is no box with more than one coin as shown in Figure 1 that shows the work of the machine in distributing six pennies by always selecting a pair of pennies in the leftmost box with at least two coins. It’s required to design an algorithm using greedy method automate the machine.

Figure 1

## Detailed Solution

Pseudo-code

ALGORITHM pennyMachine(n)

//INPUT: n - number of pennies

//OUTPUT: machine’s distribution of n pennies

result[i] <- n

Do

result[i+1] <- (result[i] / 2)

result[i] <- (result[i] % 2)

i++

While result[i] > 1

return result

C++ Code with detailed steps’ explanation

#include <iostream>

#include <vector>

using namespace std;

vector<int> pennyMachine(int n)

{/\* n represents the number of pennies \*/

int index=0;

vector<int> result;

/\* Put the number of in the first box \*/

result.push\_back(n);

/\* Iterate on each box if it's greater than 1

\* and Divide the value in the box by 2.

\* Put the remainder in the same box

\* and the division result in the next box \*/

do{

result.push\_back(result.at(index) / 2);

result.at(index) = result.at(index) % 2;

index++;

}while(result.at(index) > 1);

/\* Return the result boxes \*/

return result;

}

int main()

{

/\* Ask the user for the number of pennies \*/

int pennies=0;

cout << "Enter number of pennies: ";

cin >> pennies;

vector<int> boxes = pennyMachine(pennies);

/\* Print the final boxes distribution \*/

int length = boxes.size();

for(int j=0; j<length; j++)

{

cout << boxes.at(j) << " ";

}

}

## Complexity Analysis

* Time Complexity of the algorithm is **O(log(n))**  
  which is shown from the division of the number of pennies by 2 each time to get the final result.
* Space Complexity of the algorithm is also **O(log(n))**which is shown from the number of boxes used to store the final values of the distribution of pennies.

## Comparison With Another Algorithm

Another way to solve this problem can be reached using a brute force algorithm as follows:

ALGORITHM pennyMachine(n)

//INPUT: n - number of pennies

//OUTPUT: machine’s distribution of n pennies

result[i] <- n

While result[i] > 1

result[i] <- (result[i] - 2)

nextIndex <- (nextIndex + 1)

if result[i] <= 1

result[i+1] <- (nextIndex)

nextIndex <- 0

i <- (i + 1)

return result

But this algorithm is slower than the greedy one as its time complexity is **O(n)**.

## Sample Output

Sample output of randomly selected input:

![Text

Description automatically generated with medium confidence](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAeAB4AAD/4RDcRXhpZgAATU0AKgAAAAgABAE7AAIAAAAGAAAISodpAAQAAAABAAAIUJydAAEAAAAMAAAQyOocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEFzcmFuAAAFkAMAAgAAABQAABCekAQAAgAAABQAABCykpEAAgAAAAMxNQAAkpIAAgAAAAMxNQAA6hwABwAACAwAAAiSAAAAABzqAAAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAMjAyMjowNToxNiAwNTozNTo1MwAyMDIyOjA1OjE2IDA1OjM1OjUzAAAAQQBzAHIAYQBuAAAA/+ELGGh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8APD94cGFja2V0IGJlZ2luPSfvu78nIGlkPSdXNU0wTXBDZWhpSHpyZVN6TlRjemtjOWQnPz4NCjx4OnhtcG1ldGEgeG1sbnM6eD0iYWRvYmU6bnM6bWV0YS8iPjxyZGY6UkRGIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iLz48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOnhtcD0iaHR0cDovL25zLmFkb2JlLmNvbS94YXAvMS4wLyI+PHhtcDpDcmVhdGVEYXRlPjIwMjItMDUtMTZUMDU6MzU6NTMuMTUyPC94bXA6Q3JlYXRlRGF0ZT48L3JkZjpEZXNjcmlwdGlvbj48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOmRjPSJodHRwOi8vcHVybC5vcmcvZGMvZWxlbWVudHMvMS4xLyI+PGRjOmNyZWF0b3I+PHJkZjpTZXEgeG1sbnM6cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOmxpPkFzcmFuPC9yZGY6bGk+PC9yZGY6U2VxPg0KCQkJPC9kYzpjcmVhdG9yPjwvcmRmOkRlc2NyaXB0aW9uPjwvcmRmOlJERj48L3g6eG1wbWV0YT4NCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgPD94cGFja2V0IGVuZD0ndyc/Pv/bAEMABwUFBgUEBwYFBggHBwgKEQsKCQkKFQ8QDBEYFRoZGBUYFxseJyEbHSUdFxgiLiIlKCkrLCsaIC8zLyoyJyorKv/bAEMBBwgICgkKFAsLFCocGBwqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKv/AABEIAHYBjwMBIgACEQEDEQH/xAAfAAABBQEBAQEBAQAAAAAAAAAAAQIDBAUGBwgJCgv/xAC1EAACAQMDAgQDBQUEBAAAAX0BAgMABBEFEiExQQYTUWEHInEUMoGRoQgjQrHBFVLR8CQzYnKCCQoWFxgZGiUmJygpKjQ1Njc4OTpDREVGR0hJSlNUVVZXWFlaY2RlZmdoaWpzdHV2d3h5eoOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4eLj5OXm5+jp6vHy8/T19vf4+fr/xAAfAQADAQEBAQEBAQEBAAAAAAAAAQIDBAUGBwgJCgv/xAC1EQACAQIEBAMEBwUEBAABAncAAQIDEQQFITEGEkFRB2FxEyIygQgUQpGhscEJIzNS8BVictEKFiQ04SXxFxgZGiYnKCkqNTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqCg4SFhoeIiYqSk5SVlpeYmZqio6Slpqeoqaqys7S1tre4ubrCw8TFxsfIycrS09TV1tfY2dri4+Tl5ufo6ery8/T19vf4+fr/2gAMAwEAAhEDEQA/APdbL/j71D/r4H/oqOpZ7y2tWAubiGElWcCRwuVUZY89gOp7VWijeZtUijme3d5dqyxhS0ZMKfMNwIyOvII9jWL4b0O+0TVJTqFtbahNcITJraOwnk5yEkRyxVfQI5TOcIg4oAtaL438Pa7o1zqllqtoLS0d1uJHuI8QhWK7nIYhVO3cCTyCDV1PEeiSaWmpR6xp7WEhYJdLdIYmKgkgPnBwFbPPY+ledXFjqUvhuxt1h1i0GkeIJ7i+W2sd8pjaWcxyRCSNkmALo+EDkDoNwFXotCiuTYXVv/bWpLc+IIrm6k1SwEDHZAy+Z5Yij2rwo3MoyR1PFG7+78ba/j+Abfj+F9Pw/E7OTxX4diMAl17TENzt8gNeRjzdwBXbz82QQRjrkVJFr+l3i366bqen3U9gD9pjS7U+QwzxIRkpyp6jjB44rkde0SS6j+IbjTZJpL3TUigPkFjPtt2wqcfMQx6DvUmqaPLFqEYsdPkWJfC91afuoTgNmLZHwOv3sL9cd6lvR/P8m/0/H76S2/rql+v4fd0sviXStOsIZ9d1XS9Pd4kkcPfL5Y3dNrtt3KSDg4GcdKm1LxFomjeR/bGsWFh9oz5P2q6SLzen3dxGeo6eorl9L0Vn8Qw3F5pzHZ4ZhtQ8sJwGLNvjyR14XK/Subt/7dTwtpWk3lvq1kn/AAj8MSix0kTTXM+GVoJXkjdIlA28OFyWJ3cEVUtLv+t5f5fiTHW3y/T/AD/A9BufGGh2niy38O3OoW8eo3ERlSNp0HdQEwWzubdkDHIB9Kuw69pFzq8ulW+q2UuowgmWzjuEaaMccsgOR1HUdxXGaPDfafd+EGubS9jLeHmsmlFrJJ5E58ghZMA7Puty2Bwcmszwfoc6R6Jp+q3niVdU0p3kME2nRLaRz7HVpPtCwDerbmPEpZt43c5w2raev5v9LfeiU7q/p+X+eh2mpeNNJs7+KwtL2yvb83kVrNaRXaebBvcLuZBkjGRwQPrW9HNFK8iRSI7RNskVWBKNgHB9Dgg/QivLbSGQ+HPCOht4f1D+09I1O2a8kNhII4WViJJllI2uGJJyhJw+TjnHV61oWo6x4gWewih0drfYv9so5a5mUcmNY1wpT5jgyllDA/ujw1C2+f4WX/BH9r5fqzqWYIpZyFVRkkngCsa18TWf9gDWtYlttJsJXzBNdXIRXjJ+R2LYCluoXJ4I5zwLmrWk914evbO3kJnltZIkdiASxQgHgYHPtXA6mLm+HhHW7WfVrTTLezlhlOlWaXM1rOyoo3RNFI2BskjJVSyk4OAWpdX8v1/y/Efb5/p/n+B6Db6tp128CWl/aztcQ/aIVimVjLFkDeuDyvI5HHIqhP4n0+ENdm/0xtLjtWuHuRfLuGG28LjaUzkbtw+YYx6cLqnhu6vfDlhpnhYasL15bu9/tDUIBbMkcm8SRnEahDKWwF2hlB3kAqK39Oge81yG6stKubG0k8OiGKGa3aLyW3nERBAAYDt6c9KmTdtPP8m/zVv6Q1a+v9apflr/AJnQaH4l0rxD4fi1rTbyF7N4xI7+ap8n5QxVyCQrKDyM8Vc/tGyxbH7Zb4uxm3Pmr++G3dlOfm+XnjtzWN4GfzfAWk21xa3NvLbWcVtPDd2zwsrqgDDDgZGf4hkHsTXLaVpuu2cGp40xp38OafPp+kRXCEpeFiWU843Dy1gTIPXeMirnZSaW39f8CxMLuKv/AF/WtzsV8WaNeaTqF9omp2OrCwhaWVLO6STBCkgEqTtzg9auadqtvf2sD+ZFHPJax3L2/mAtGjjgkdcZBGcc4Neb2ttqV/rd7dj+2r5JPDlzAZrzShZqJSyEQxp5aPgc4D7uvDEhq1YJLrQdXhuLjStRnW88PW1rCLW0eX9/GXJjfAxGT5i4Z9q8HJGDUvRN/wBfa/OyGtf6/wAP5Xf3HXXviTQ9OsLe+1DWdPtLS6AMFxPdIkcuRkbWJw3HPFXY7y2lumtormF51jWVolcFgjZCsR1wcHB74NeS6bpeqaYNDutVn8Qabbt4ctbRDpmmJdNDKuTLFJG0ErpnKHIUA7CCchRXe+FbOx0uK30yz02+ja2023UXt5bqrSR5bbEXUDLJg5UABdwx1q2km15/5i/r8v8AOxW1nxndafcao9jpC3un6KFOpXBufLdMqHYRJsIkKxkMcsvXAya2bHWVu9UuLGSIRukaXEDq+5Z4W4DjgYIIIK84yvPzCuP1YX+nr4u0dNNvLm412Qvp0sNq7wsZIEiIkkUFY9rKSd5X5cYzW5Z22zxpZQQtvGl6QYZ3HdpHj2A574hc47ZGeoqY62v/AE7Nv7nZfMctNv61Vvv1E8UeItc0G6tBZaNp95bXl1FaRSTak8LiR+PmUQMAox1DE+1aFj4jsZb1NJ1DUNKh13GZdNt79ZXQ4zwCFc/Lg8qP61S8aWtxdR6F9lglm8rWrWWTy0LbEDHLHHQDua47UzrWqaxbJcQaqk1t4hik+w22lBLSOBZ/lnM7RkyMUwSUk43HKgBqIauz72/L/Nilpr5f5/5Hd6l4s0mxj1OKC/srrUdOtZLmXT0uk84BF3cqMso6ckdxV211ezuGt4WuYI7ua2F0LUyjzBHxltvUqCcZxivPpYb1E1/S9FtL+506e11CWSK90ySKS1uH3HEMpUCZHZmwo3Edd2MCm6V4b1ltHu/Dt7K9tqV95dxLrMllJNHe24/5d32snlFR+72bhlSSvLNhK7v6L9f667FS0+9/pb+vM7Hwr4si8VPqjW1q0NvZXQhhlZs/aUMauJQMcKQ3HXIwe+K0ZNe0iHWI9Jm1Wyj1KQbksmuEEzjBOQmdx4B7dq5Hw6mueH9Z8R3OrxC9t5r62jiTTdLeIuTFDH5ibpWHlqOGxnG1jn+Gsay0Oc6jeaXr974lWabW2vFitdNie1mHnCSKT7R5B2hVCA7pAw2EAY2g1o2vl+n+d/60T0Tf9f10PSl1jTXgimTUbRoppvs8UgnUrJLkrsU55bIIwOcg0yHXtIudXl0q31Wyl1GEEy2cdwjTRjjlkByOo6juK4Ww0++Xx8+tDT7uLSr2eeGyhaBs2lyUUNdsm3KrJsZcnp1/5anFXwXokkZ0Gw1m78Tf2jpMrSPbS6dElqkwV1eQXCwDer7mIxKWbeN3OcKOtv6/q2/zWwS0v/X9dvv8jvNa1e70++0q0sLOG6l1C4MRM1wYVjVUZ2bIRsnC8DjPqKoP4k1bUL68h8L6NbX1vYyGCa6vL82ySSj7yRhY5C208EnaM8DODibXf+Rp8Mf9fc//AKTSVj6FqDeDY73RdW07U3C3k9za3VlYTXUdxHNK8gGYlbYylipD7egIJBzR1DodHqWtHT5tLt1tjJc6jcCFYi4XywFLuxIB+6qn6nAyM5rMsfEusayReaHodvPo5l2JdXF/5Us6BsNJFGI2BXrt3OhbHQAglLOP/hIPFmo3zqyQWFt/Z0GSDiRwHnPHGR+6X6q1UvCerv4f0HT/AA1q2laouoadHHZK0FhLLBcBQFWVZlUxqrDB+dlK8ggYpr+v6+5/PyE9v6/ry+Xmdk88Mc0cUkqJJLny0ZgC+Bk4HfAonnhtYWmuZUhiX7zyMFUfUmuV13wxq+oeKtOvrTxDqltbxvKWWFLTbbZjwNu+Escnrkt17VP408P6nrXhx7bTdXvoZljCmKFbfFycry5eM4PBPylRSGdPXOa14vi0fxNpuktatMl0VFzcB8C0DsUhLDHO9wV6jFbGm2c9jZiG61G51KQMT590sSufbEaIuB9K4CTw54m8TW3iK8jvbTTU1WYxQQ3ulyNcRxwkrEwbzk25YGQZQ43Z56Ur2Y90d3qOu6To6s2rapZWIVQ7G6uEjwpOAfmI4J4+tLPrelWtnHd3Wp2cNtLGZY5pLhFR0AyWDE4IwQc9MGuBttdU+NdI1nXNEv0um0DEgjsJJpbWUy4YeWilwCQRuAxjGeDVOCD+xpvCFxqujXHli91G8is44DNJZRyFnj/dqCcqrKCqglSTgcU+l/63a/4Iuv3flf8A4B3114ms0s9Mu9Mkg1K11G8S1jnt5wyfNu+YMMhsFcY/WrMerxR21/dajcWFta2czI063YZUVQMmQlVEbAk5XJxxzzxxNvp95dTHUoNNure0vfE8V3DBJAyOkSxBGleMgGPc6s3zAHBBOCabd6dfQhb2fTrq4sbPxRLe3NtHAzvJFtISVU6uFco+FyflyASKUfP/AIb4fyu/+B0H5f18X+SOyfxb4cj0qPVJPEGlrp8shjju2vYxE7jPyh84J4PGe1V9U8b+HtHu9Lgv9VtI/wC1Tm3kNxGEK7SwcksPlOMAjOSQO9Ymua1c3Nxps2mwarpum3Hn+ffW+hySXiyDYEURvGxjVvmy7xkEIBxkGsnRLW+0jw94PmvtN1JVstVvDcR/ZDJLCri4VC0cS9CWXlF2jIIwKpIT0O/j1eKO2v7rUbiwtrWzmZGnW7DKiqBkyEqojYEnK5OOOeeFtNf0e/006hY6tY3NkH8s3MNyjxhsgbdwOM5IGPcVw9zp1/DtvZtOubiytPE817c2yQF3ki2MElVOrhXKuAuSduQCQKj13T5PEcfiG/s9Ku/sF/Hp9uYZ7OSKS6aOcmRzE6h9oRlXJAyFPYA1K1in5L8l/X9aU9G/66tf8E6+98T28vh2/wBT8MT6drTWIYyIl8AgKjLKXRXw2O2PTp1rXs7j7XYwXG3Z50avtznGRnGa4vVLOa38Q+Lp0tXitJvD8QEojwjyL9oBAOMEhSvHYY9q67SP+QHY/wDXvH/6CKa1v8v1/wAhS0aXr+j/AFC0hmmutQW2ljiYXAJMkZcY8qPsGFWfsGo/8/1r/wCAjf8Axym6R/yENR/67j/0VHXn8Xin4neIfFPiKz8JReEksdHvzZg6mtyJW+UNn5CQevt9KV9bf1/WodLnoX2DUf8An+tf/ARv/jlH2DUf+f61/wDARv8A45TLG/u9M8NQXXja70uzvFAW6mt5DHbBi2FCmTnnKjnuaki8SaHPrD6RBrWny6kmQ1kl0hmXHXKA7hj6VVtbB0uJ9g1H/n+tf/ARv/jlH2DUf+f61/8AARv/AI5XLeOfihpfhTUdKsINU0h7q41SG1v4p7tQ9pCwJaRlDArgY5bjkV1mkeItE19ZToOsWGpiEgSmyukm2Z6Z2k4zg9fSktVdf1/Vwejsxn2DUf8An+tf/ARv/jlH2DUf+f61/wDARv8A45SXXinw/Y6oum3uu6bbX74C2s15GkrZ6YQnP6VdvL+z06OOTULuC1SSRYkaeQIGdjhVBPUk9B1NAbFP7BqP/P8AWv8A4CN/8co+waj/AM/1r/4CN/8AHKdb+JNDvNXk0q01nT59RiBMlnFdI0yY65QHI6+laVAGX9g1H/n+tf8AwEb/AOOUfYNR/wCf61/8BG/+OVfubmCytZbm8njt7eFC8ksrhURQMkkngAetZx8VeHl1CCwbXtMF5cqrQWxvI/MlDDKlVzkgjkY60AO+waj/AM/1r/4CN/8AHKig0W5tfN+zTWEPnSGWTy7Ervc9WOJOScDmtCbUbK3vreyuLy3iu7oMYLd5VWSYKMttUnLYHXHShtRsl1JdOa8txetEZltjKvmGMHBcLnO3PGelAFT7BqP/AD/Wv/gI3/xyj7BqP/P9a/8AgI3/AMcog8T6BcywxW2uabNJPE00SR3cbGSNc7nUA8qMHJHAwa5X4j+M2svhXeeIPBmr20zpNCkV5atHcJzMqsAfmU8Ej2o3dv63sM6r7BqP/P8AWv8A4CN/8co+waj/AM/1r/4CN/8AHKWDxJok+rnSItZ0+TU1Hz2S3SGZcDJzGDuH5VY1PV9N0W0N1rGoWun24ODNdTLEmfTcxAoEtSt9g1H/AJ/rX/wEb/45R9g1H/n+tf8AwEb/AOOVb0/UrHVrNbvSr23vbZ/uzW0qyI30ZSRVC58YeGrOzju7vxFpMFtK7RxzS30ao7KcMoYtgkEEEdsUASfYNR/5/rX/AMBG/wDjlH2DUf8An+tf/ARv/jlaSOskavGyujAFWU5BB7g06gDL+waj/wA/1r/4CN/8cqOHR7q3Mpt57GIzOZJClkV3sQAWOJOTgDn2qfVvEOjaAkb67q9hpiynEbXlykIc+g3EZq5bXMF5bR3FnPHcQSDcksThlceoI4NAFD7BqP8Az/Wv/gI3/wAco+waj/z/AFr/AOAjf/HK4S81Dxn4z8e6/o/hfxLB4YsNAaGJpP7OS7luZJE3kkOcKo6DHNeiaXb3lppNrb6nff2heRRKs135Ii85wOX2DhcnnA6ULVXDrYrfYNR/5/rX/wABG/8AjlH2DUf+f61/8BG/+OVqVj3Xi7w3Y25nvfEOlW0Ima3Mk17GiiVfvJkt94dx1FAEn2DUf+f61/8AARv/AI5R9g1H/n+tf/ARv/jlaMU8U9uk8MqSQuodJEYFWUjIIPQj3rO0/wAUaBq19JZaXrmm3t3Hnfb213HJIuOuVUkijrYOlw+waj/z/Wv/AICN/wDHKPsGo/8AP9a/+Ajf/HKlu9d0mwmmiv8AVLK2kghFxMk1wiGOInaHYE8LnjJ4zT9N1jTNZsjeaPqNpf2oJUz2s6ypkdRuUkUAZl/4a/tUQjVBpd6LeQSwi407zPLcdGXc5wR6jmrf2DUf+f61/wDARv8A45UMvjLwxBHavP4j0iNLwZtme+iAnGcZQlvm5GOO9XtT1fTdFsjeaxqFrp9tuC+fdzLEmT0G5iBmjZAUoNHurVGS1nsYVZ2kZY7IqCzHLMcSdSSST3JqX7BqP/P9a/8AgI3/AMcrG+HV3q99oN1ca54k0jxEzXknkXWklTEkXG1CV43D05IyOT1rd1bX9G0GNJNc1ax01JDhGvLlIQx9AWIzRsC12I/sGo/8/wBa/wDgI3/xyj7BqP8Az/Wv/gI3/wAcq/bXMF5bR3FnPHcQSDcksThlceoI4NZyeK/D0mr/ANlR69pj6ju2/Y1vIzNn02Z3Z/CjrYOlx32DUf8An+tf/ARv/jlH2DUf+f61/wDARv8A45WjLLHBC8s8ixxxqWd3OFUDkkk9BUdle2uo2cV5p9zDd20y7o54JA6OPUMOCKAMn/hH7j+1f7S+02v2vyfI8z7M/wBzduxjzcde+M0T+H7i4v7W9mubVri03+S/2Zxs3jDcebg5A71u1W1C/h0zT5ry6JEUK7mwMk+w9zS0SFKSinKRU+waj/z/AFr/AOAjf/HKPsGo/wDP9a/+Ajf/AByqtlc+Jry5hmltNPs7JyC0Uju04X8PlzW9TMqVZVVdJpeasZf2DUf+f61/8BG/+OUfYNR/5/rX/wABG/8AjlalFBsZf2DUf+f61/8AARv/AI5R9g1H/n+tf/ARv/jlalFAGFqHh19Xsms9W/s2+tXILQXOn+YjEHIyrORwasJpt/HGqR3loqKMKq2bAAen+srVooAydI/5CGo/9dx/6Kjrybw74F/4Svx147uf+Ep8S6L5OttH5Wj6j9nR/wB2p3MNpye2fSvWdI/5CGo/9dx/6KjrmdY+Cnw/1/WbrVdX0D7Re3chkml+23C72PfCyAD8BSt71/L9V/kP7NvP/P8AzMj4waWdJ+Al3pov768MD20f2u8m8y4f/SE+ZnwMt74rM8e+BvDfg/SPCFz4f0m3tL238QWSC9RB9okDMd2+T7z575zXozeBfDjeDY/CjadnRItuy18+Tja+8fPu3fe561e1nw/pniCC1h1e2+0R2lzHdQjzGXZKhyrfKRnHoeKpaSv/AHk/krf5MneNn2a+881+Mfh3w8mo+FNSu9H0xZLrxHbR3t1LaxgzRlWBWRyPmXgcE44FehQaBpfhzSr9vCui6fp88kLMFs7VIRK4U7d20DPJ/WrOu+H9K8T6TJpmv2MV9ZyEFopRxkdCCOQfcc1m+EvAHhnwKl0nhXTfsC3ZUzDz5Jd5XOPvscdT0qbPlcdv+GS/Qq/vKX9bv/M8H8GeCfE3jHwCLmy0r4c6mNQEjXF7qEdy+oCVidxkkAykgPpgDtXa/EDQ9Q034I+GND8QXn2i9h1Kwtri4t5W+b95tyrkBs4xzgGus1n4K/D3X9Ul1HU/DcL3Ux3SPDPLCHJ6krG6jJ7nGTW/c+D9Cu/D+n6JcWO/TtNaF7WHznHlmLHl/MGycYHUnPfNPdfNP7mJ7u3Z/iYw+EvhC3u9HutJ0uPSbnSLlbiGexREklwCNkrlSzqc85Ofeu0oooFY5j4lf8kt8T/9gu4/9FmvJ/EPgHw1pX7NB1a30uF9WNhbXp1KVd1z5rGMkiT7wHOMA4x2r3XU9OtdY0u603UYvOtLuJoZo9xXejDBGQQRwexqle+FtG1Hwp/wjV5Z+ZpHkJb/AGbzXH7tMbRuB3cbRznNLa7W+n4X/wCAXdXjfZX/ABt/kzh/FFysfxj+Gst1IF86C+QM3AZzCuB9TTX1Sxvv2mobeyuop5bPw7LHcLG2fKczKdp9Dgg4967XxJ4M8P8Ai/SI9M8R6ZHfWkTBo0ZmVkIGMhlIYcccHmqugfDrwn4XvILrQNFhsZ4IHt0kjd8lGILbsn5iSo+ZsnjGcU3r/wCTfin/AJkWfLb0/Bo+ffDfg3Rr7wd8PJrm3LT6zr08V7MGxJJCDIvk7uojIXBUcfM3rWxrVlb2fwp+JlhYxJaWsXihI4ooFCLEvmQDCgcCvbbPwD4asLHSLS003y4NGuGubBPPkPkyMWLNktlvvNw2RzUkngbw3Np+rWM+lxy2us3Bub6KV2cTSnHzcn5T8o+7jGOKb3b/AK3j/k/vLbvK/wDX2v8ANHmnxG8HeFfCHhPQ5PC+nWdnr8eqWg02ePH2m4kMq7sv96TIJJySP0qt49sdb8QfHYabaW3he9+zaSkljaeJlleFg7sJGjROGfKgHOeAMV6F4b+Evgfwlqo1LQPD8NveqCEmeWSZkzx8vmM204JGRitLxT4H8N+NraKDxRpMN+sJJjZiyOmeuHUhgD6A80n09W/vViV19Evudzkvhj4F8S+FPE2s6hrUfh6xs9Rij22GgecsCyqT8/lyDCkg4+U49qw/hF8P/C+u+D73Utb0a11K7uL+7hMl4gm8tBMwAjDZCdScrg5Jr0Pwj8PfC/gQ3Z8K6Z9g+2bPP/0iWXftzt++xxjcenrWpoegaZ4b042Gi232a2MrzFPMZ/ndizHLEnkk0NJvytb8V/wRa2+d/wAGReF/D1t4T8M2Wh2E9xPbWSeXE9y4aTbkkAkADjOBx0ArWoopttu7BJJWR5H4U0Xw94s+JXjaXxdZWeq6xZ6gLeC2v0WUQWgRTGUjfIAJJOQOp96i8B+GtA1u68f+FoIhe+Df7QgFvbxzt5SybA8qIytkAOF4BxXb+Kvhl4O8a3kd34l0OG8uY12iZZHicjsCyMpI+ua3NG0TTfD2lxabollDY2cP3IYV2gep9yfU8mlFJKz7W/LX8PvY5NuV13v/AF/Wx4h4P+DngPVPH3jbTb/QvNtNLu7eOzj+2Tr5SvDuYZDgnJ55Jr3eztILCxgs7RPLt7eNYokyTtVRgDJ5PA71U0/QNM0vVdS1KwtvKu9UkSS8k8xm81kXapwSQMDjgCtGnfRLyX32FbVv1/MK8V+HHgvwrq9z4z1nxNp1pfyx69ewbtQAkjt4g27Kq3ypySS3X3r2qvFfB3wd8N6/f+JNU8aeGppL5tfumge5eeESQ7gUYKGCspyecEGpS975P80U/h+a/JnPvd3tt+zJqkekTqul/wBtyWtrNLI3liwa528sPm2HJUkc7Savx/C3xlfyaPd6fp3w50pLO5iuYdQ0JLiKcqpGQH2kOCMj5sg5/GvcP7J07+x/7K+w239neV5P2Tyl8ry8Y2bMYxjtXGWXwQ+HenaxBqll4dWG8t51uInW7nwjq24ELv28EdMYqou0r+mvpb/K+/VktXjb1/H+rfJGBrfhfSvFP7SKQa9bC8tbXw/HcC2kOYpHE7Ab16MBuJweKs+DNMstC+Lfj/TdGtYrGx+yWUwtbdAkauY3yQo4GfavQxoGmL4mbxALb/iaNaizNx5jcxBt23bnb15zjPvSW/h7S7XXNQ1iC123+pRxx3U3mMfMVAQowTgYBPQCot7vKuz/ABb/AOAVvJv0/C3+TPEvAfw/8MXf7PN3rOo6Tb32o3VhdSG6ulEkkRTzAgjY8xgbR93Fel+BtOsfEXwf8Mwa/ZW2pwtptuzR3kKzKSEGCQwIz71u6f4W0fSvC3/COWFn5WleU8P2fzXb5HzuG4ktzuPfvVHVfh54X1vwpZeGtU0vz9IsCht7b7RKuzYpVfmVgxwCRyTVyesrbO34X/zElt8/xt/kc38EIIrbwzr0FtEkMMXiO+SOONQqoocAAAcAAdq4e+0jxF4o+Mni1LWw8F6nc2ckMMVv4njllkit/LDKYkUbdpJJJxnJPNeleH/g54E8La5b6xoWhfZb+23eVN9snfbuUqeGcg8EjkVf8VfDbwj42njn8TaJDezxDaswd4pMehZCCR7E4pPeL7K35a/h+ILqu7v+Ldjnfh74L1rwh4X8R2niW90vTra+lknhXSJZBBYKyYcp5oBQA/NjJA9q4v8AsG0+Gnhyzn8W+DfDeu6JaSxBPEOlBY7xRuGyV1YbnYtjOxz+NeveHvAnhvwroN1ouh6WkGm3bs89tJI8yyFlCtnzC3BUAY6Vjab8Fvh7pGsJqdj4Zt1uo33oZJZJFVs5BCMxUEHpxx2pq3Pfpp+H9bh9mz8/x/rsddqUFjfaPdQaosb2E0DLcCU4QxlfmyewxnmqfhXT9C0vwxZWnhIW40eND9mNtL5sbKSSSHyd2SSc5NaF9Y22p6fcWN/Cs9rcxtFNE3R0YYIP4Gq2haFpvhnRLfSNDtVtLG2UrFCrFtuSSeSSSSSTkml3F2NCud8dI7eFJijFAksTMwGdoDjnHfHWuipskaTRNHKivG4KsrDIIPYik1cyr0vbUZ0tuZNferHGarbXWg2lvqtl4gvryZ5o1EE8weO4DEZCrjjjnjNdrWPYeE9D0y8F1Zaekcw+6xZm2/QEkD8K2KfQ5sJh6lJyctL20TbWl7u7tvf8AooooO8KKKKACiiigDJ0j/kIaj/13H/oqOsbxH8VfBPhLVRpmv8AiCC2veN0KxvKyZ6b9inZwQfmxxzWzpPF/qP/AF2H/oqOvFPh74paTw1d2HhXSY9c8Za5eXM+pyXKfuLRWldVNxJ/d2jiMEk5OOvK1vZf1/XfoPS12e92d5bahZQ3djPHcW06CSKaJgyupGQQR1FTV5j8CILnTvB2raLdTpcDSdburOOSNAiMAQTtUdBuZjjtmvTqp26eT+/UlX1T6X/AxfEnjDw94Qs/tXiTVrbT4yCVWV8u/wDuoMs34A1o6bqNrq+l2upadL51pdwrNBJtK70YZU4IBHB6EVx/xO8O6PN4O8R67NptvJqsWi3EMd28YLomxzhSen3jyOea0/ht/wAkt8L/APYJtv8A0UtKOqd+lvxv/kN6W87/AIW/zOmqtqOp2OkWL3mq3lvZWsf35riURov1J4qzWdrPh7SPEUMEOu6db38VvMJ4o7hN6q4BAODweCetLXoBW8MeLtD8ZWE974av1vraCdreSRY3UBwASBuAzww5GRz1rarzL4KgLF41VQAB4qvAAO3CV6bVO1k11Sf3q4bNrs3+DCiiikAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAZOkf8hDUf+u4/wDRUdcEvgbx34T1DVbf4danoEWj6pcvdldUgk86zkf73llBtcdwH6cD1J73SP8AkIaj/wBdx/6KjrWo63H0Of8ABPhOLwZ4Yi0tLl72cyPPdXci7WuJnO53IycZPbJ4A610FFFNu5K0OG+I+keOtf0+40jwk/h5NMvrN7e6bUzOJgWyCUKArjaR1B5p/wAN9J8a6Do8OkeL30F7KwtIreybSzMZDsG3MhkAHQDoBzmu2opLT+v67jeoVjeKP+Em/sgf8IX/AGT/AGj5gz/a3meTs5z/AKv5t3THbrWzRSeoHlPw88JfEnwnrN2NTl8Ky6XqmpS6hf8A2drkzq0gGRFkBQMgcNnvzXq1FFV0sHW4UUUUgCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDJ0j/kIaj/ANdx/wCio61qydI/5CGo/wDXcf8AoqOtagAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDJ0j/kIaj/ANdx/wCio61qydI/5CGo/wDXcf8AoqOtagAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDJ0j/kIaj/ANdx/wCio61qKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAP/9k=)

Figure 2

![A picture containing table

Description automatically generated](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAeAB4AAD/4RDcRXhpZgAATU0AKgAAAAgABAE7AAIAAAAGAAAISodpAAQAAAABAAAIUJydAAEAAAAMAAAQyOocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEFzcmFuAAAFkAMAAgAAABQAABCekAQAAgAAABQAABCykpEAAgAAAAMzNgAAkpIAAgAAAAMzNgAA6hwABwAACAwAAAiSAAAAABzqAAAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAMjAyMjowNToxNiAwNTozNjo0NAAyMDIyOjA1OjE2IDA1OjM2OjQ0AAAAQQBzAHIAYQBuAAAA/+ELGGh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8APD94cGFja2V0IGJlZ2luPSfvu78nIGlkPSdXNU0wTXBDZWhpSHpyZVN6TlRjemtjOWQnPz4NCjx4OnhtcG1ldGEgeG1sbnM6eD0iYWRvYmU6bnM6bWV0YS8iPjxyZGY6UkRGIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iLz48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOnhtcD0iaHR0cDovL25zLmFkb2JlLmNvbS94YXAvMS4wLyI+PHhtcDpDcmVhdGVEYXRlPjIwMjItMDUtMTZUMDU6MzY6NDQuMzU5PC94bXA6Q3JlYXRlRGF0ZT48L3JkZjpEZXNjcmlwdGlvbj48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOmRjPSJodHRwOi8vcHVybC5vcmcvZGMvZWxlbWVudHMvMS4xLyI+PGRjOmNyZWF0b3I+PHJkZjpTZXEgeG1sbnM6cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOmxpPkFzcmFuPC9yZGY6bGk+PC9yZGY6U2VxPg0KCQkJPC9kYzpjcmVhdG9yPjwvcmRmOkRlc2NyaXB0aW9uPjwvcmRmOlJERj48L3g6eG1wbWV0YT4NCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgPD94cGFja2V0IGVuZD0ndyc/Pv/bAEMABwUFBgUEBwYFBggHBwgKEQsKCQkKFQ8QDBEYFRoZGBUYFxseJyEbHSUdFxgiLiIlKCkrLCsaIC8zLyoyJyorKv/bAEMBBwgICgkKFAsLFCocGBwqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKv/AABEIAHQBjQMBIgACEQEDEQH/xAAfAAABBQEBAQEBAQAAAAAAAAAAAQIDBAUGBwgJCgv/xAC1EAACAQMDAgQDBQUEBAAAAX0BAgMABBEFEiExQQYTUWEHInEUMoGRoQgjQrHBFVLR8CQzYnKCCQoWFxgZGiUmJygpKjQ1Njc4OTpDREVGR0hJSlNUVVZXWFlaY2RlZmdoaWpzdHV2d3h5eoOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4eLj5OXm5+jp6vHy8/T19vf4+fr/xAAfAQADAQEBAQEBAQEBAAAAAAAAAQIDBAUGBwgJCgv/xAC1EQACAQIEBAMEBwUEBAABAncAAQIDEQQFITEGEkFRB2FxEyIygQgUQpGhscEJIzNS8BVictEKFiQ04SXxFxgZGiYnKCkqNTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqCg4SFhoeIiYqSk5SVlpeYmZqio6Slpqeoqaqys7S1tre4ubrCw8TFxsfIycrS09TV1tfY2dri4+Tl5ufo6ery8/T19vf4+fr/2gAMAwEAAhEDEQA/APdY/wDkNXP/AF7w/wDoUlW6px/8hq5/694v/QpK57TfEd5rPiQWc8kehiFmI026j/0y7Cjlhn5PLB7xmQHj51OVo62Dpc62ivLBe6n4WtfF+oWl/e6lcya1FaQwSx2+0SSrbqJOFTJAfABYKcDOOWq3PqvjHS9F1SSYajDGv2X7Jdawlk0okeYJIm22JUptIPIBGW5PGBapP0/JO34h1a8/1sekUVxNxB4gfxPHoUHim8hiTS2uXu/stsZnl80hesewKBxjbnAHOck0/Dms+IJv+ES1PU9W+0pr8bLcWS28aQxHyWkV4yBvB+TBDMwO44A4AFqr/wBbtfoD0/r5nd2d7a6hbC4sLmG6gLMolgkDqSCQRkcZBBB9xU9ee2Or6/rMfhe3j1h7M6jFePdzxW8TSMI2AXZuUqpGe6kYzxnBEcHiXUr7QbGyOo6pJrP2q7gI0m0tvOukt5TGZCZh5MY+6TnbkkBfQpO+w7dT0aivLIJ9V8TL4HvrzVryyuvt11BKII7f5mjjnXeQUcBiEwQpK8nHYiza+IfF2tajLf6TY6qbaLU3tUtwLEWbQxzGNy5Z/P34VmyMAHaNrDJara2Jvpf+uv8Akei3FzBZ273F3NHBDGMvJK4VVHuTwKlryzxJcatrvgPX9Z/tp4LaO6mtk03yI/J8qKfyvmJXzPMO0tkOACQNpHXrvFGuz6BdQTwXlrM0qkLpEiMZ7naeWg8sM5IByRscYA5QZal2H1a7HS1FHdW8tzNbxTxPPBt82JXBaPcMruHUZxxnrSWk73NnFPLazWjyKGaCYqXjPodrMufoSK89fXb+w8H6M2kw3E2oeItRk86W1EJlQkSSNt81hHuCxhBuOAB0OACbOwbq6PSKgmvbW3ube3uLmGKe5YrBE8gVpSBkhQeWIAycdq4KXWvFWh+HZ9V1Zb8RaffIUt7xLQ3Oo27hVMZEBKeYHY7Nu3dhVI5JN3StX1e6bw0bjVA41uK5uZPISMrEpjDRoh2nITPU5yc544B0bD+v6/rqdvRXH/DOzntPC0hn1K6vQ99dBROsQEeLiQEjYi/ePJznnpgcVVsvEOqTanaeH5L3/iY2+p3AvpvLTJtIx5inG3A3LLApIH8Rwc80dbB0ud1UVvcwXdus9pNHPC/3ZInDK3OOCK8+0fxNqk3izRmhutVvdG1l5ljmvra2hgcCNpFaBVxOANuP3oIIOc9CZfBmo3UK+FdPjl22tzZX8sse0fMyTRhTnGRje3T1o1ul3DrY9BorzE+LtfvjYaZYjUJ57q51J5LjTUtRMsVvcmNEXzyI+jLkkMcDpklh1XhlPElxb2lxr80lqYRPFJazRQmS4G8eVK7RMVVggOVU7SWJwMAULX+v6/ryB6Oxranrmk6IsTazqllp6zNtiN3cJEHPoNxGT9KspcwS3EkEc0bzRBTJGrgsgbO0kdRnBx64rmI4Le9+JOuR6hGsqx6PbRIkoyvlSPN5mAeMEqoP+6Kx/CbyQ6L4BugztLcW72bsxO6SDyWkUnPXBiQ57ZOOpoX9fj/l/VtR7/15f1/w52Gq+JtB0KaOLW9b07TpJF3Il5dxxFx6gMRkVZ0zVtO1q0+1aPqFrqFvuKedazLKm4dRuUkZ5rA+Jv8AyTfVv9xP/Ri1h+NvE+p6fcand6Bd6rN/Y6obmCG2thZxNgPtleXEjllYf6liVBHGeot7AejVHb3MF5bpcWk0c8MgykkThlYeoI4NchLqep2/j0QavqF9YWM8yx6bHFbRPaXY8rLJJJtMiS7t2MsgIVdu45rmPA/im+bwyZIEult/D9mAdKt7fzLnUncZV8bSwiycKV5JDEnC4Kv1GeqLeWz3klmlxE1zEiySQBwXRWyFYr1AODg98Gpq8x8DXkP/AAsjUvOGoSajf6bby3Us+mXMCmUSTZA8yNdqKu1VJxkL3bNT6xr/AIr1HxBrdv4btdVxpMiwW62i2Jhnl8pZD5/nuJNp3gfu9vGTuJPyvt/XWwur/rpc9HorgL/xBrcHjSDw19t8ltVMVzHc7I2NggVjJB0wzMYzsLA5BkPOwZqWviHxdrWoy3+k2Oqm2i1N7VLcCxFm0McxjcuWfz9+FZsjAB2jawyWdtQ2X9ep6BqGp2GkWTXmrXtvY2qEBp7mVY0Uk4GWYgcmodS1/R9GtYrnV9WsbCCY4ilurlIlkOM8FiAeOeKz/GH/AB46b/2FrP8A9HrVHw2kd3478VXl6qPf2lxFaQFgN0VqYY3AXuAzs5PqR/sjCWqf9dv8wejt/XX/ACOl/tGy/s0aj9st/sJjEouvNXyihGQ2/OMY5z0qrqniXQtDkiTW9a0/TnmBMa3d0kRcDqQGIzWX4hhS+1TRvDtuiJDJP9uu0QAAQwsGAx/tSmMe43VX8AxRXdnrGo3Ucb6ldapdQ3rsoLbY5WSKM/7IjC4H+0T3NG/9f1/SDZf15/5HXg5GRyKK5TxXr+paHLp1vpWh39xC91DGZbY2wjZSSDEBJKrA4HXAHvV7Wdav7Pwy1/baHqD3LRMTbo9t5lsQpO590oQgY/hZvpSbSi5dhpNtRN2oLm9tbIRG8uYbfzpFii82QLvc9FGepPYDmszwrq17rGg21xqOm3djKYYyWuWhPnEqCXURO2BnscH2rjfHd3aa14ln0e4XUimm2DSxSWWnXFyI7yTBicmJG2sgTcM4/wBZTl7rs/6/r8xR95XR6ZRXBjxJqXiKPwkdJv20pdXjn+2DyEaVGSP5lUOCFdXBHII4OQarT67r8EculQ6uklza+IINOGozQIWlikjWTDooC7wH25ULkgcDnJtLl/rdL9RJ3jzf1tc9AuLmC0gae7mjgiXG6SVwqjJwMk+5psd7azXk1pFcwvc26q00CyAvGGztLL1AODjPXBrzvxFdaja2PiPQb/U5NVjt7eyu4rmeONJU8ycqY28tVU8x7gdoPzY5xmtPXPFGqadc+IIrWRC8dzYWlmZYwUga4KoXbGCwBbdgntjil1S7j62/rv8AkdxRXI6ub7QNIW3n8Saxe3d9cpFaGKztmumfaWMaYjWIAhSd0gwoByRwRyt5dazr3hgQajqOo2Vxp3ia2tQ7x2vnupkhZTJsV4967+NmAcDI6imtXZeX5pfqJ6K7/rRv9D1GO9tZrya0iuYXubdVaaBZAXjDZ2ll6gHBxnrg1PXCar4k1TSpNctorrzpIJ9PsrSWeJSInn2oZXChd2C27HA4wMCk1a/1rwvcXdo2uXGqCbRru8hlu4IBJbSwBMEeWiKVPmDhlPK9cHFJtJN9r/gr/kUk20u5291dW9jaS3V7PFb28Kl5JpnCIijqSTwB70lpd21/aRXVjcRXNvMoaOaFw6OPUMOCK5DRL/WrTxTpVlq2rNqKatpT3bI1vHGtvLGY8+XtAOwiU8MWPyjnrWt4G/5EjTP+uR/9CNU007P+rOxKd9TR8wRatdOyuwFvDwiFz96TsASaWWe1nMZns7iQxuHTfYynYw6MMrweTzU1n/yMM/8A17xf+hS1i+Jfi34J8H60+k+Ita+x3yIrtF9kmkwrDIOUQj9am6GJNoWg3E2pSXGmXkq6qu29heC5MU3AXcY8bA2FUbgA3A5psHh/Q4LOe1Nrq1xHO0bSG7e9uGPltuQBpCzABucAgdfWt3wv4s0Xxno/9qeGr37bZ+Y0Xm+U8fzDGRhwD3HatinawXuc3tsTqh1H7JefazB9n8z7LP8A6vduxjbjr361Wt9N0e0t9Mht9PvEj0n/AI8l+z3B8r5CncfN8rEc561v61reneHtJl1PWbpbWzh2h5WBOCSFAwASckgcVeByMihbaf1/VwOTtNN0ixaya10+8Q2CSJbf6PcHYJCC45HOSB1z7VSl8MeHpVjA0/U4WimmmWS2+2QuGlbfIN6EMVZuSpO3gccCu5ooA4n/AIRvw8ul2enw6XfW9tYztcWq20V1C0DsWyVZMMAd7DGcYOMY4pT4f0I6w+ppY6nFPJMs8iQi7jhllGMO8K4jZvlHJUngeldrRR1uBwd94S8M6lcXUt5peoSC7fzJ4Qt2sMj8fOYhhN/yj5tu7IBzmt4T2q3L3C2dwJ3RUeUWEu5lUkgE7ckAs2B7n1reoo6WAxft8f8AzwvP/AOX/wCJrn/7A06WzvNPvobu502a6+128P2O4SS1kLb2KSL8w+cllK7SuSAcYx3VFAHGW2h6LbfZiLPVJmtbg3MbXP2udvNK7d7GTJYheBuzt7YqS00nRbCaCW0069ja3lmmh/cXBEbSnMmARgAnt0HYCuvqjrmrwaBoF9q94kj29jbvcSLEAXKqpJABIGePUUm7K7BK7sjD0uy0zRWuTptpqEK3UrTSRmK5dA7EsxVGBVMliSFABJzSx2elQ61c6tHp90L66iWGab7JMd6LnAxtx364ycDPQVvaTqUOs6LZanaq6Q3kCTxrIAGCuoYA4JGcH1q3VNWdmJNNXXU4Ww8LeHNMvrW7tNN1ETWZY2plW7kW2DKVKxq2QikHG1QF4HHAxJP4d0C4sLO0On6jFHYszWz263cMsRbO4CRMPg5ORnB49BXbUUhnDHwt4aGl22nw6Te21vZzST232WK6geB3LFykiYZQdzZAIGDjGOKt2ml6RY3FpcW9jf8AnWcUsUMskVzIwWRgz7iwJYkqCS2T7111FAHHazoui6/Kkup2GoNIkbReZBHdQM8bdY2Me0shx91sj2qRbWH+24b50uvKtIDDZ2yafKqw7sbm6YJwoA4G0ZHOTXW0UbAc1qcdhrOmzWGpWd5NbTACSP7LOu7BBHIUHqBWVqfhXw3rF1dT6jpmoS/bAPtEQW7SKYgABmiXCFwAMMRuG0YPAq94t+Iml+EtQt9NksdU1fVLiIzR6fpFobifygcFyuQAueOtbHh3W/8AhItDh1L+zNR0vzSw+y6nb+TOmCR8yZOM4yOelC11QPTRmENG0Ya1/ahs9Te53+YFcXbQq+3bvEJ/dhscbgueTzzSQaJodrLp0ttp19DJpsbRWzxw3KlUbqjED51JAO1sjIBxkV2VFAHJX+m6PqbXzXun3kpv7QWVz/o9wPMhBYheBx99uRg89elVb7w5oGoXpu5rDU45miWKRrb7ZAJ0XO1ZBHtEgGSMPngkdDXb0UBv/X9djjptF0O4W6E2nXrNdXMd3LJ5NzvMsZBRg2Mrt2jABAHIxyajPh/QjrD6mljqcU8kyzyJCLuOGWUYw7wriNm+UclSeB6V2tFGwHH+IreXWNNjhsZJ7S4huYbmOWbTJpkDRuGAKDaSDjH3hTNS0TSNYuob2/tdSS+ii8r7VZLd2kjJnJUtEVYrnkKSQDXZ0UB1ucbpOm22landXca3jiSKKCBDYzEwxICdpYglyXZ2LHk5Gc4zUd/4d0LUdSl1CWz1WC6nVVnksmvLXzwvC+YIiofA4+bPHHStXw/4007xJ4g1zSLGC8juNEnEFy08GxHY55Q55HHfHY9CDXQ0b2YbXRz7SWbxxRvYzskLK0atYSEIV6EfJwR2qR7uGSNkktrp0YEMrWUpBB7Eba3KKNw2MOO6ghiSOK2uo40UKqLYygKB0AG3gVVsodP06e8mtLO8SS9m8+4c207F32hc8g4GFAAHAx0rpqKPMDzfUfCGl3erafJb2V3DZw3Nzdzqkd1HIJpQPnjZRlDnJO0r1PqafqvhXTbnRbDSrCxuEs4dRW8uBNBcs8vLFmMhUu0hJzuJzkdRgV6LQSACScAdSaVl+X4f8MH9fecVb+H9Bt9Ou7JdP1GSK9kWS5edLqaWZlxt3SvlzjaMDPGKs3Wm6NepqKXmmXE6amipdpJZzMsoUYHBXAwPTHrW3a6/pd7qT2Fnexz3CJvZY8sAOn3hx36ZpdW13TtDjjfVLjyFlJCHYzZI/wB0Gh2a1Mfb0lF1FJWW7urL5nKnwxoDWQtpLbWpAJhMk0k9888bgFQUmLeYvBIwrAYJ9TTl8M+HE0i50xNLvktrm4W6lCxXQczDaRIJPvB8op3Agk89SSek0rxJpOtzPFpd357xruYeW64HT+ICtSn/AF/X3FU6tOrHmpyTXk7nItpOiy297BcaZdXEd/GkV0txazy+cqLtXduByQO/Unk881Ba+H9CtLe7iWy1ScXkJt5pLs3lxIYiCCgeQsyryeAQM89a7Wijc02OSubO0MkF3Y208OoWds9tZzzWNxIkSttyCg27h8i9weOozS+H4Ronh+z06U3Vy9vHtaVdPmQOc5JC4OOvTJrrKKAMez/5GGf/AK94v/Qpa8xvtZ8SaP8AHzxK3hXwr/wkckmm2Ymj/tGO08kDdg5cHdnnp6V6dZ/8jDP/ANe8X/oUtcXrPhb4gWXxL1TxL4Jl8NGDUbSC3ePV2uNy+WDyBGMdT6mp+1F+v5Mf2WvT80bGpeIvEcfwo1vWtS0f/hHNYtbO4kit/tUd35ZVCVfcBtPrgiuBv9d+KWn/AA5h+IN34h0+GCKCC4bQ1sEdZ4mKjc833g7BtxCgAdBivQZ9I8W6/wDDTWdI8UPoqazfW89vE2nGUW4DJhSxcFs5JzgGq3iDwTqWrfBP/hD7ea1XUP7PgtfNkdhFuTZk5Ck4+U44qtm2vL9b/oCs+VPbW/4f8E5P496b4gvPA9xqtp4m+zaKVtQ+kHT433uZlw/nE7hglTgcfL713PhLw94v0m+km8T+N/8AhIbV4dsdt/ZMNr5bZB3bkJJ4yMe9TeNfCDeLvh7deHRdC0mlij8ufbuCSIysDjuMr+RqDwhH8Q4r14/HEvhuaySHbFJpYnEzyAjlw/y4xnp3xTVk2l3f5f1+BGrjFs8n1j4zvqev6kIviNa+ELeyunt7ayGgvfPcKhxvkk2kLuIOAvQHnmurh+Jerax+zvqfi20lhttWs0kjFxBHmN3RwvmKrg8MDnDDjNXV8I/EHwlqOoJ8P9S0C40i+upLxbTW45g1q8h3MqNF1UnJ56fqdjxB4e8T+K/hHqGh61JpMWv30DRs9o0otQd+VwWBf7oGeDzms3f2Ttvb8f6+Rrp7Rdr/AIf18zBeD4utZ2fiK21nTrlmaJ38NwWcYjeJiN3+kuQ28KcnGBkceh9UqvYQNa6bbW8hBeGFEYr0JAA4qxWsrXaW1zKN7JvcZOxS3kZTgqhIP4V4XoOsfFbxN8Mj4uXxTY6XHZ20ssUH9nRTNqHllizSNgCPO3aAo6DJ55r3SZDJBIi9WUgZ+lcT4W8FajonwZ/4RK7mtXv/ALHcQeZE7GLdIXKnJUHHzDPFZSulJrfp+JpG10n319DM1bxzq83hX4eatYyrZvr+pWcd7GkasrJIhLoNwOBkdRz71seLtf1PS/iB4K02xufKtNUublLuPy1bzVSHcoyQSMHngisrU/hvq138LPDWi2WoW1rrvh1ra4tpzuaBpoVxhuN2w5POM+3aqtv4H8e6t498PeJ/F+s6O40qWXOnackqQxo0ZXcpcFnckjO7AAHGc1s7c7ttzfhp/wAEzV+W/wDd/HX/AIByeqfEH4h2x1/V7K/tX07RPErabFZvbx7r4NKFETPgeWqqVww5JY5PFadxrPik6D8S/Dni/VLfVJtP0YXEc0FqsAi86Fy0YA+8q4ABPJ79eNW6+Futz+HNf09LqwEupeKf7ZhYyPtWHzEba3ycPhTwMj3rXv8A4fX+oeIPHF093bRW3iTTIbK3YbmeJljZGZlwBjLDGCfwrFp+z5fL/wBtX63Nrr2l+l//AG7/ACOLtZfibovwqsfFsPiDTbay03TYZ10EWSus1uiDl5z8wcrzhcDPetv4lfFddDt/D1rYavBoH9uW5u31O5tHuvssO0EBYlB3OS3GeBg5qJPAvxM1XwzB4Q17WPDtl4eWBLWefTIpmu5oFAGz958gLAYJHT0PSum8W+CNSubrRdX8EX9vper6JE9vbpdRl7eeBgoMTgcgfKCCOn6jWbu79L/Pr+tvx+eUE0kutvl0/wCD+Gpx/wAMvihcav49HhqTxZb+MLW4tHnj1BNLawlgdCMoyEBSpB4I545q7Y6h8Q/F/inxVpWjeI7PRtP0vVGhjvXsUuJ1GxSIlQ4Xb1JZiT83HSut8K/8LG/tST/hN/8AhF/sHknZ/ZH2jzfMyMZ8zjbjd75xT/B3ha98Pa54pvL2W3ePWNTN5biJmJVNirhsgYOR2z9anqvR/fdfoHT5r7rMseB28Uf8I+0HjhIjqVvcSQrcxbALuIH5JtqkhSw/h4xjoK6Oiine4bHnXiDV/FfiTx5d+FPBeqW2gw6XbRT3+qS2ouZN8mSkaRt8uMLkk+vHTlnhzXfGVl4k1nwXr13Yarq8Gm/b9L1RoDBHOCSgE0afdw+M7e2aseJPB3ii28aS+K/h9qWmw3t5bpbX9lqyObecJ9xwyfMGAJHvVvwb4Q1mx1+/8T+M9Rtb7Xb6FLYJZRFLe0hUlvLj3fMcscknnpSitNfO/wCNrfh8ipPXTyt+F7/ieb2v/C1P+F36h5X/AAh/9uf2HF5u77V9m8jzTt2/xb92c54xXt2h/wBr/wBiW3/CSfYv7U2n7R/Z+/yN2TjZv+bGMde+awrbwtew/F298UtLbmxuNIjsVjDN5okWUuSRjG3B9c+1dZTXwJev5sm3vN+n5IK8lk1P4g+JfiZ4p8O+HtetdH03TJLdhey2aTywh4Q3lohADbmySWJxxivWq8TtR46Pxm8ef8IG3h8J5lmLn+2POznyPlKeX+Oc+1StZW8n+aK+zc3dB8f6rovhbxgvjKWLUtS8IyESXNvGIheI0YeI7RwrHOCBwK81l+N2o2lkNbT4labf3gCyP4bXw/LHCem6NbgruyBnBJxn2r17Rfht5fhLxBYeJtROpal4lLvqd5FGIwCybFWMdlQdM/XvgZdnpfxl0W0i06x1HwdqdrbKI4rrUI7mOeRAMDeE+XOPTNPXm17L7+v9WBfD8393T+rieN/Fniv/AISbwdp/gmS3hbX7a4d0vEUomERhIxxuJQMxwCMnGc1N4W1PxdofxRbwl4s1+HxDBdaWdQt7tbFLV4mWQIUKpwRznPJ+lb2t+Fr7U/iN4W8QQy2622kR3S3CMzB2MqBV2DGDyOckfjSz+F72X4uWnilZbcWMOkSWLRlm80yNKHBAxjbgeufamrJrzv8Arb9CdXH7v0ucLoV98TPHF54gj0zxPaaHYaXq11bQXR0+K4ln2t8se0gKqKuPm5Yk966nwTreveO/hJDd/wBoRaVrkolt2vobdZUSSOQpvEbcHIXoeOa0PAXha98LW+ux6hLbyHUdZub+LyGY7Y5CNobIHzcc4yPes/wt4X8UeDvhc2kaTLpEuuxzyywtctK1rh5i/wAxUB/uE9B19qlfDZ9l9/Up6yuu7+7X/gFf4Zan4kl8R+LtE8Va7/bj6PcwRQ3P2OO2yHj3n5UHuOpPSsH4ofFJtE8ZR+GLfxTb+FEjtVuLjUpdNe+kZmPyxpGAVHAyS3qMVY8LeFviponjPUdXvJfB7wa1dwS6isTXRZERQh8oEAA7c/eJ59q3vFPhHxKvi9fFfgHUtPttRktVs7uz1SN2t7lFYsrEp8ysMnp19uct/Zf3/d/mKOnN+H3/AORi/CL4j3HivWdX0O51u38RpYRRzwavDZNaGZWJBV4mAwwPccYNYVr4x8V674ivYLj4g6b4Q1S3unji8N6hpKAOgYiPM0hBfeMHKZ68V6d4S/4Tj/S/+E7/AOEf/g+y/wBjef77t/m/8Bxj3ri9f8G/E3xNps+ga5e+Db/Spg0f2+exmN2in+NYwfLDjtg9e9D3XoC2aPVYvO+zJ5xQT7BvKA7d2OcZ5xmsPwZZ+KbLRpY/G+p2epX5uZGjltIvLVYT91SMDkc9vbJ6nS0bTBovh+x0uKeScWVtHbrNNyz7FChm9zis3wZZ+KbLRpY/G+p2epX5uZGjltIvLVYT91SMDkc9vbJ6mnbmdv61Er8qudBTJoY7iCSGdQ8cilXU9weop9Ml8zyX8jb5u07N/wB3PbOO1Q9gexzFvaW9j8RYbezhSGFNLIVEXAH7yrfiO31E3+l3ul2P25rSR2eLzlj4K46t9azTpPi466NW36J54g8jbmbbtznOMZz+Nb2pf255MH9kf2f5uP332rftzgfd2/j1o+yv66s8alHmpVYuEoq91ZK/S1t10K+leIpLzUm03U9Ol06+CeYsbOHV19Qw61t1h6Zot6NWOr65dRTXnleVHFbqViiUnJxnkn3NblPod+EdZwftb76Xte3nbS4UUUUHWFFFFAGPZ/8AIwz/APXvF/6FLWxWPZ/8jDP/ANe8X/oUteafG+Lxhp/hjWNdtfFk2nabamAWdlpqeTIzM6KzSzZ3d2wq4HTPudUu40rnsdFMiJMKE8naKfQSndXCiuZ8Y6V4j1aO2h0DX10S0Ac3ssVuJLh1wNqxs3Cd8tjI4xWN8D9RvtV+D+j3eq3txfXTtOHuLmVpJHxO4GWYkngAfhQtU/Ib0sd/RRWb4hg1i50K4h8N3lvZak4AhuLmLzEj5GTt7nGcds4zSYGlRXlnwwfWbX4leONH1vxBf639h+xFJbt8KGeNmcpGPljBPZR2HXrXqdVbRPuHVoKKQ52nbgHHGRXjckfjDRfjN4Ki8R+LJ9Q/tRr0zWVovkWiKkOVUIPv4JzufJ6enKWsrA9m+x7LRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAVnWOgaZpusalqtlbeXe6oY2u5fMY+aUXavBOBgccAVo0UAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQBj2f/Iwz/8AXvF/6FLXln7Qnjzw3H4J1fwi+pY1xvs7i08iTp5iPnft2/dGetep2f8AyMM//XvF/wChS1sUtbplRdnc57wf418P+NtMkuvDGofbobZxFK3kyR7XwDjDqCeD2roaKKp7kRVlY5rxp478OeCLGJ/FGo/YVvA6QHyJJN5A5HyKcdR1rz79nnxv4euvA2l+FINQ3a3AtxNJa+RINqeczZ3ldp4Ze/evZqKUdL36/wBfqOWqSCqGua5p3hvRLnV9auPs1jaqGml2M+0EgdFBJ5I6Cr9FJ36B6ngfgj4reDI/jH4wvX1nFvrktjFp7/ZZv37KhQjGzK/MQPmxXvlFFV0S7A7uTfcK8A8XfFvwRd/F/wAGapb63vs9HN6t9L9kmHkl4tq8FMtk8fKDXv8ARSV1JMfRruU9I1ay13R7XVNKm8+yu4xLDLsZd6nocMAR+Iq5RRR6CCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAMez/wCRhn/694v/AEKWtisez/5GGf8A694v/Qpa2KACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAMez/5GGf/AK94v/Qpa2Kx7P8A5GGf/r3i/wDQpa2KACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAMez/wCRhn/694v/AEKWtiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigD//2Q==)

Figure 3

![Text

Description automatically generated with medium confidence](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAeAB4AAD/4RDcRXhpZgAATU0AKgAAAAgABAE7AAIAAAAGAAAISodpAAQAAAABAAAIUJydAAEAAAAMAAAQyOocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEFzcmFuAAAFkAMAAgAAABQAABCekAQAAgAAABQAABCykpEAAgAAAAM2MAAAkpIAAgAAAAM2MAAA6hwABwAACAwAAAiSAAAAABzqAAAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAMjAyMjowNToxNiAwNTozNzo0NAAyMDIyOjA1OjE2IDA1OjM3OjQ0AAAAQQBzAHIAYQBuAAAA/+ELGGh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8APD94cGFja2V0IGJlZ2luPSfvu78nIGlkPSdXNU0wTXBDZWhpSHpyZVN6TlRjemtjOWQnPz4NCjx4OnhtcG1ldGEgeG1sbnM6eD0iYWRvYmU6bnM6bWV0YS8iPjxyZGY6UkRGIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iLz48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOnhtcD0iaHR0cDovL25zLmFkb2JlLmNvbS94YXAvMS4wLyI+PHhtcDpDcmVhdGVEYXRlPjIwMjItMDUtMTZUMDU6Mzc6NDQuNjAzPC94bXA6Q3JlYXRlRGF0ZT48L3JkZjpEZXNjcmlwdGlvbj48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOmRjPSJodHRwOi8vcHVybC5vcmcvZGMvZWxlbWVudHMvMS4xLyI+PGRjOmNyZWF0b3I+PHJkZjpTZXEgeG1sbnM6cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOmxpPkFzcmFuPC9yZGY6bGk+PC9yZGY6U2VxPg0KCQkJPC9kYzpjcmVhdG9yPjwvcmRmOkRlc2NyaXB0aW9uPjwvcmRmOlJERj48L3g6eG1wbWV0YT4NCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgPD94cGFja2V0IGVuZD0ndyc/Pv/bAEMABwUFBgUEBwYFBggHBwgKEQsKCQkKFQ8QDBEYFRoZGBUYFxseJyEbHSUdFxgiLiIlKCkrLCsaIC8zLyoyJyorKv/bAEMBBwgICgkKFAsLFCocGBwqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKv/AABEIAG4BkQMBIgACEQEDEQH/xAAfAAABBQEBAQEBAQAAAAAAAAAAAQIDBAUGBwgJCgv/xAC1EAACAQMDAgQDBQUEBAAAAX0BAgMABBEFEiExQQYTUWEHInEUMoGRoQgjQrHBFVLR8CQzYnKCCQoWFxgZGiUmJygpKjQ1Njc4OTpDREVGR0hJSlNUVVZXWFlaY2RlZmdoaWpzdHV2d3h5eoOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4eLj5OXm5+jp6vHy8/T19vf4+fr/xAAfAQADAQEBAQEBAQEBAAAAAAAAAQIDBAUGBwgJCgv/xAC1EQACAQIEBAMEBwUEBAABAncAAQIDEQQFITEGEkFRB2FxEyIygQgUQpGhscEJIzNS8BVictEKFiQ04SXxFxgZGiYnKCkqNTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqCg4SFhoeIiYqSk5SVlpeYmZqio6Slpqeoqaqys7S1tre4ubrCw8TFxsfIycrS09TV1tfY2dri4+Tl5ufo6ery8/T19vf4+fr/2gAMAwEAAhEDEQA/APdZP+Q1bf8AXvN/6FHVuqkn/Iatv+veb/0KOuZ1Cy1ZvEXm+IIJ9T0gTL9lj019qQ85DXEP3pMHAyGdTgN5aYzR1Dpc6iLUbWbUrjT45d11bRpJLHtPyq+7ac4wc7G6elWa8v1uztLDxr4tk0WzsYfE1xoqTaY3kos8su2cO0Zxlm4XOP8AZz2rI0yy0s6Pqt5o3iDQriQaDdC6tNJ094JpcoMNdFp3PmKc/fAfLPz96pcrRb7K/wCf+Q0ryS8/8v8AM9norziTwtop1DwdYvp1u9rPHcT3MTRgrdSGBctKP+WhPBO7OSB6VlWOg6bp+i2upWlqqXtn4pNra3OSZIIPtpi8lGPKx7CRsHy8nitLe/yedvxS/Ui94c/9bN/oeq21xJO04ltJrcRyFEaUoRKMD512seOcfNg8HjpU9eX3NlDqAjtbpS8MnjKQOgYjePLclTjqD0I7gkHg1Tu5dO0W313Q2sdKi0b+344IxqIxYWCtapKWdOFK784TKgs45FRF80b+n5R/+SLas2vX83/kep32o2umwxy3svlJLNHAh2k5d2CqOB3JAz0qzXicVnpNx4GvP7Ui0y70zTfFEHlyfYvKtYIGeAuY0dn8uI7mP3tpDZ+6ak162sb/AMSavBq2t+HtKi2QjQ3vdPaWRLcxLseykE6ANv3cRgtkLnjaKpbf12T+/X8H2J6/13a/T+rnsd1cLaWc1xIGKQxtIwXqQBnio9Ovo9T0u1v4FdYrqFJkDgBgrKCM47815jra6B/aPieLxu0EmrpCo0lpx+/eH7OuGtR94OZRIT5fzZxntW8kLz+CfDKWNlqE2p/2fF9muLGYQi3/AHaZaSRvl2Z2koVk3YyI328JbP5fjf8Ay+QdUvX9DuapQ6pFcaxcadDHK7W0atNMAPLRm5CZzndj5sAYAIyRkZTRo9Tj0qFddmt5r7H7xrZCqD25+8R/ewueu1eg4TXJb9PBfilNOdoruPWf9NdY2kZLdnjLNsQhiPII4BBIBwQeh1t/W6X6jWq/rz/yPSaguriS3WMxWk10XkVGWEoCgJ5c7mXgdTjJ9Aa8mgsNAh0DW5rHXNEvYZUt1trXw1YiCOO9DMYHQedIpmLbemDhRu4q7p5drWyOreX/AMJJ/wAJLCNX2jo+GMYTPPleWV2e2c87qOqXmvxaX6/l30Tel/X8Ff8Ar5npVpqNrfTXUVrL5j2c3kTjaRsfarY5HPDKcjjmrNcN4IsdB07xh4rt7K00+01MX+4xxRJHN5DRQkHAAOwuSc9N2e9UdXtJB4xufCkcRNpr91DqUhx8oiQf6Qp9iYogf+u9C15fNL+vz+4ffyb/AK/I9Hqjpmqwar9r+zpIv2S6e1feAMsmMkYJ45ryfUr+zuvGNvqFqmjWGqReIIraSCG3aTUzGJvKMkswcGON16IyFcMoDZIxrzXMOnvFqN7IsFlZ+MJ3ubiQ4SFWikQM7dFXc6jJ45FEdbf11j/mEtG1/W0v8j0+ivGtV1H+049YubG7to9Dn8TRi/urq2ea3aD7FEFZ1DJuhaTZltwUggklc56DwnZeHbRbVm1fTNSil1Ytpa6RbPDaW9x9nIZYwsjrgrvY87dxbjdTSv8Ah+Nv8wen9ev+R1+ta/Fo81rbJZ3WoXt4W8iztNnmOFGWbLsqhRkZJYckDkkVFpviuw1RdPaFLiJL/wAxIzMgXZMmd8DDORINr8Yx8hwazPEF/baH480jVtYnitNNawubU3c77I4pWeJ1DMeF3BGwSRkrjrisexjdvC9tMiGN9U8TC8sVIwWia58zdjAPMSu/0NKOrX9W1t+WoS0T/rpe/wCh2WvazJodg12mk3upRxqzyizaEGJVGSx8yRM/hk+1N8Pa5Jr+nrenSb7ToZUSSE3jQkyowyGAjkfHGPvYPNP8S/8AIp6t/wBeU3/oBry3W7jTb3RdG0zU4NDheHw9BNb3GrWzXUs7MhBitYgyHzBtUllYtkoNvQib25vK36/5Dtdr5/p/mey1R0vVYNXiuHtkkUW9zLbP5gAy0bbSRgnjI4ryyS5sHk8Malqkmm67qMum2C/2Vft/piFjn7Tak5y+TlgAM+WCXXHL9Pd4fFF5J4hSBtKGrXX9jGZ9tst/5r4884OGPAjOCAd38ZWratNx9fwa/wA/v0JTvHm9PxT/AK9NT0Z/E1gnjCLw0PMe/ktWuiVUFI0VgMMc8Md2QMdB24zsV5dpdl4i0fx9oQ1ex0tria2vpry7i1GR2mJa33vtNuuNoCKqZxtAG4Y5q+Kb6w1/xPbXWqapoNv4eutLjm0yXxBp7TW0rszeYV3Sxqku0x9cttJxj5srov67/wCRXmet0V49rMP2O3sY7HUJdQii0qFfE15HGytLYblKMQWLCTb5nUlvLMh67ah1+1sr3xJrFvrGs+H9JjCQrojX+ntLJHb+Uu17JxOgDb93EYLZC542in1/r+v6fYX9f1/XbuetazqkWiaLd6ncRyyxWsTStHCAXYDsMkDP1IrPvvFCWIsoDpWoT6neIZE0yARNMijG5nbf5SgZAyXwSQASeKh8aI8fw51VJZPNdbJgzkY3HHJx2rOvtQtfDHxIutT8QTpZ6fqWnwW9vfTnbDHJE8rNGznhCwkBGcBsEDJFHWwldq/9f0joE1yOPQLjVtUs7rS47VHknhuwm+NUySfkZlPAyNrH8+KpXHi5EuIbSx0fU9Qv3t0uJrO2WINao3TzHeRYwxORtDEnBIBAzVLWdSsvFa6Tpek3Ud5Z6hcmaeaA743t4CGcBhwys/locZBDN6VUstY07wl4v8RR+J72DTBqd0l5Z3d5II4p4xDHGUEjcblZD8mc4YEDBpev9f1+g/T+v+G/VHZWVybyxiuGt5rZpFBaGdQHjPdWwSMj2JHoSOanrkPGi6rrvg95vDF5avbTxBtj2Mk7zgspUoVkTAxz0bI5FbYt9f8A7GaNtT006luyLkadJ5IXPTyvP3Zx33/h2oA1KyvEniG08L6HLqd/HNLHGQoit0DSSEnoqkjJxk/QGsrwDZa7ZaDGmt3Nq0eZNkEdg8EiHzGOWLStkHqBtHUcmqXiC41PVPH9lY6JZWN+miwG8uY7u7aBVllDRx/MscmSE804IH3gc0nfSw13Z2dvPFdW0VxbuskMqB43U5DKRkEfhUlePT3aW/hOz0HxIotm0PX7eCdLWV2VLZsvEQ4CsUEZ2k4HCNkCk1YaP/YPjFvCIjXw59mtB/xKseQ115h8ww7ON2zys7O+O4NO6abX9aJ/qLbT+t2tPuv6Hrd9dx2Gn3F5MGaO3iaVwg5IUEnHvxUNnqRvfszw2dx9nubYXC3DFAi5xhCN27dg54BHB5rzl/8AhH31bUm8Cm1az/sG6OqvYEGJpTjyfMI6zf64nPz4zu6ip7ltSWORtBDHUh4NX7KEGW8zPy4989PelfX+u0v/AJEdtv66x/zPTKrajqFrpWm3F/fy+VbW6GSWTaW2qOpwASfwrzj7d4F0/wAF3lx4WtNJvpWt4Yb9S+xtjuql7xgCwxlmfzBuOHz3rnxY6fc+F/HtlbrpGo6fDYQXcC6Vp5isxLslzJFGXkUt8gBdDjK+oNVbf+v6/pCWtj2n7RJ9uWAWkxiaIv8AaQU8sHIGwjduyc5+7jjrnip681vmRZ75vAAtTjwvIdOGnhfL3+a2PL2/Lndnp3pukJ4VOv8Ah5vAItPtqu/9qG1/13keS+77X/Fv8zy/9Z827PfdS6ff+bX6C6f12T/XQ9LYlVJCliBkAdTWdoesrrljJcCzubJ4p3gkgudm9HQ4PKMyn8Ca838OaVZaVZ+ANTsrdY7+9upILm65Mk0TQTNsZupUFFIU8LtGMV3fhP8A1Grf9ha5/wDQ6Fvbyf5r/MctLf13/wAjfooooApzBzrNuIwpf7PNtDHAJ3R9Tzip/J1X/n2s/wDwKb/43Uf/ADH7T/rhL/6FHXPeMfH2taD4vsPDvhvwp/wkN7eWj3e3+0Utdiq20/fUg9R3pX1sHS50vk6r/wA+1n/4FN/8bo8nVf8An2s//Apv/jdVPCWs+JNYhuW8VeFf+EceNlEKf2jHd+cCDk5QDbjjr1zXQ1WwtzJ8nVf+faz/APApv/jdHk6r/wA+1n/4FN/8bqXxDrCeH/DWo6xLE0yWFtJcNGpwXCKWwD2zinaFqf8Abfh3TtUEXki9tY7jyt27ZvUNjOBnGeuKW9/6/rYZB5Oq/wDPtZ/+BTf/ABujydV/59rP/wACm/8Ajda1FAGT5Oq/8+1n/wCBTf8AxujydV/59rP/AMCm/wDjda1FAGT5Oq/8+1n/AOBTf/G6PJ1X/n2s/wDwKb/43WtRQBk+Tqv/AD7Wf/gU3/xuqy6Tdpqj6illaLcyRCF3F7IA6gkjK7MEjJwcZGSM81v0UAZPk6r/AM+1n/4FN/8AG6PJ1X/n2s//AAKb/wCN1rUyZzHBI69VUkZ+lJuyuxrV2MzydV/59rP/AMCm/wDjdHk6r/z7Wf8A4FN/8brN+G/ia88Y/D7TNd1OKCK6uw5dLdSqDbIyjAJJ6KO9dRVNNOzJTuZPk6r/AM+1n/4FN/8AG6PJ1X/n2s//AAKb/wCN1rUUhmT5Oq/8+1n/AOBTf/G6PJ1X/n2s/wDwKb/43WtRQBk+Tqv/AD7Wf/gU3/xuq8mlXkuoRX0llaNcQoyRsbyQhA3UhdmAeMZxnHGcVvUUAZPk6r/z7Wf/AIFN/wDG6PJ1X/n2s/8AwKb/AON1g+OPFus6VrOjeHfCVlZ3Wt6x5rxPqEjLbwRxKCzPt+ZuoAA5rX8Kf8JZ/Z03/Ccf2N9s8391/Y/m+X5eB97zOd2c9OMYoWquD00J/J1X/n2s/wDwKb/43R5Oq/8APtZ/+BTf/G61qKAMnydV/wCfaz/8Cm/+N0eTqv8Az7Wf/gU3/wAbrWooAyfJ1X/n2s//AAKb/wCN0eTqv/PtZ/8AgU3/AMbrWooAwNR0m71bTbiwv7K0ktrmMxyoL2RCVPX5lQEfUHNPtLDUbK0itoYIGSJQqmbUJZXI93dCzH3JJrcooAwBpN2uqvqQsbP7W8IgMhvJD8gYsABswOSeQMnjPQVZ8nVf+faz/wDApv8A43VS2vPFTeP7y0utLs08MJaq1tfLNmZ5uNylc9OvYdByc4HQ0dA6mT5Oq/8APtZ/+BTf/G6PJ1X/AJ9rP/wKb/43WtRQBk+Tqv8Az7Wf/gU3/wAbo8nVf+faz/8AApv/AI3WtRQBzep6Jfar9j+0QW6/ZLpLqPZdsMsucA5iPHNGsaJfa3pM2n3cFukUxXc0V2wYbWDDGYj3FdJRR0AyfJ1X/n2s/wDwKb/43R5Oq/8APtZ/+BTf/G6oXfifUrNZLqfw5cLp8Zy05nTeF9fL6/rWlqeu22m6VFessk3nlVgijXLys3QAUrq1zlWLovm1ty6u6a076rVeaGeTqv8Az7Wf/gU3/wAbo8nVf+faz/8AApv/AI3Vey8RXR1KGz1rSJNNe5z9nczLKrkc7SR9047VvUzSlWhWTcOndNP7nZmT5Oq/8+1n/wCBTf8AxujydV/59rP/AMCm/wDjda1FBsZDQaoylTbWmCMHF24P5+XVLSNBuNDsja6faQiNpGlYzalNM7MxySXkVmJJ9TXSUUAZPk6r/wA+1n/4FN/8borWooAxf+Y/af8AXCX/ANCjrzzx3putar8dtBt/Dev/ANg3v9iXDfbPsaXPyiQZXY5A5459q9D/AOY/af8AXCX/ANCjrH8X/DWy8X67aaw+ua7o97aQNbxy6RdrbsUY5IJ2k9fep+0n/WzQ+jX9bou+HNH8RaPpN9H4p8Uf8JHLJlopf7PjtPKXbyuEJ3c85NeL+EPhjo2vfAk+I9envr3UIbG5k09vtLxrp4jL7RGqnH3l3EkHJNe1eFfBw8L2l3A3iHXtaF0QS+sXv2hosAjCHaMA55+gpdG8FadongL/AIRK0munsPIlg8yV1Mu2QsWOQoGfmOOKJJ2lbeyt+P8AVwi0mr7X1POvEXhrSfG37Pth4j8T2rX+rWHh1p7e6eeRWWQwhixCsAxJUH5ga2/hh8LfB2jaPoHibTdH8nV30+KVrn7VM2WkiG87S5Xnce30rtdO8M6fp3g+Dwzte506GzFkVnILSR7dpDEAckegFc74X+Fll4R1aC60rxJ4le0twyxaVcal5lmgIIx5e3oM5HPWtW1zya2f/B/O5FnyRT6f8D8rHmnii21jxz8S/Elve+Brjxfp+kTx2tva/wDCQLp8Vr8gbd5eQWZsk7s9OO1dZ4G0vxRo3w48VWXifTbjTLRUnbTLS5v0vHghMRynmqeVB6ZAPNdJ4m+F2keI9cOtQajrGhapIgjmvNFvTbSTqBgB+CDj1xnp6Vo6H4Nj0bw1e6NNrmtaxHeb99xqt358yB0ClVbaMAYyBjqTWLi/ZOPW1v6/ryNb/vFLpf8Ar+vmeX+Dvg7o3i74UaPqWqX2oS61Np6NY373MmNOIHyCKNWVcLgHB5JGc17Xp0E9rplrb3dybueGFEluCu0ysAAWxk4yecZPWqvh3Qrbwz4bsNEsHlktrGFYYnmILlR0yQAM/QCtKtpO8nbZmUVor7hXhPgn4aaN45uPFlz4rku760g8QXsVrYC4eKKBt+WlwhG5zkDJ4wOle7Vi+G/C1l4Wi1GPT5biQajfy38vnsp2ySEbguAPl44zk+9Z21b8n+aLvpZd/wBH/meLXd7cXP7JDG+uZLhre7WBZZm3MUS9Crk+ygD6Cu6+Mksa6L4WRpFDyeJLHYpblsMc49a3rP4c6Fa+AbjwfMs95pVwZDILhwXJdy+QVAwQxyOOMCsC3+Bnh5J7K5v9X8QapdWE8UtpPqF/5zW6xsGEaArtCEgZ4zgcEVad5K/dP8r/AJCaVnb+8vv2PO/iBoUWreLPipfXdzcbNHsrO6treOVkT7QYBtlO0jJUKQAePmNb/g/Rh4e+K9j5d5dXV1rPhVr7U7ieVmN1OZF+cqTgYBwAOg49a9C1H4caRqc/imWe5vVbxRBFBeBJEAjWNCimPK8HB5zuq3D4K02DxJZa2s101xZaX/ZcaM67GiyDkjbndx1BA9qztZWXa3/ksk/xt9y7FSd393/tv+T+/wAzx34ffCnQPE3wbttZ8SXV9c3QgnezkN28aacFd8GNQQvUbiWByas634p8S6r8F/AUEUN1eXmvzJb3axXgtJbpVViE85vuGTaCT3GR3rrU+BGgRxmzXX/FH9jliW0b+1WFoyk5KlAM4/HPvXYa94K0HxF4Yj0DULFV0+AJ9nSA+WbYoMI0ZH3SBwMfTpVy1Tt3Tt6b/f8A8OJaS+/8dvu/4Y8j8HeD/FGg+PdHu9A+Htx4R03zWTVf+KjS+iuIivVoy2dwOCCPyrWvPCNn4y+PHiew1i5vBpqadZPPZW87RJdHDACQrhio54BHJ9q63w98Mh4e1u31FfGni/UVt92LPUdV863fKlcMm0ZxnI56gVu2fhaysvGOo+JYpbg3mo28VvLGzL5arHnaVGM555yTSdrx8r/k/wBSddfl+Zm+B/AcPgNtUtdKvpX0e6mWWz0+Tcwsvlw4V2Ylgx57Y9+tdZRRRdsLWPMfHNoPF/xR0fwXqt3cQaHJps1/c20Exi+3OrhBEzDnaBkkA8/gKpad4bs/hx8YNC0fwfLcW+ka5aXJu9Lad5Y4miUFZl3klSScE5rtvGHgPSPGsdq2otdWl7ZOXtNQsJvJuLcnrscZ6+hBFV/Cfw40jwnqM+px3Wpavq1wgifUtXujcXHlg5CBsAAfQdhRBW38/ne9v68hy1/D5f1+p5hr/wANPI+MPhbTP+E18YSfbbW9k+1yarmeDYqnbG+35Qc8jBzgV7J4X8O/8Ixo/wBg/tjVdX/eNJ9p1a68+bnHy7sDgY4HuabfeFrLUPGGleI5pbhbzSopooURlEbCUANuGMk8cYI/GtqmtIpev5sT1k2/60QV474m8LW/jD9oN9L1K7uotO/4R2OW6traUx/a1FwwEbsOdmTkgYzgV7FXjnibwsPFf7QzWp1vWdGMPhxJRPpF39nkb9+w2lsHK89PpU2vOPz/ACZX2JfL80WPCNjD4B+I/inw34ZWZtGh0iPU4rJ5mlW3nJYFFLEkbgAcE5NebaT4c8Q+MvD0XiK4+HV9rer36NND4hXxakDoxJwY4s4QKeAhHGMV9BeEfA+keC7W4TTPtFxdXjh7y/vZjNcXTDoXc9cegAHXjk1zc3wV0iO7mk0HxH4o8OW8zmRrLR9UMEAYnJIQqcfQcelN7666b/f/AMN8vuE97af0v+H+f38z46s/EV38P/h3YeILubTtel1m1gubqJlaSKQpIpcFSQWxzkHrzU8ngrSPhz8WfBT+FPtds2rSXVvqPmXUkv2sLDuDOGJ+bdzxgZ7V6FqHgmy1TT9Atb++1Cc6Dcw3UE8kytLPJEpUGVip3Zzk4wSfSrOr+F7LWvEGiaxdS3CXGiyyS26xsoRy6bDvBBJGOmCKq9pN93+Fl/wSbe7byt89f+AeW2vgXTvHHxm8dQeIZ7qbSraW0ZtNimaKOeRoBh3KEMdoBwM963vhJYRRaT4v8MO80+lafrVxZW0U0rMY4Ci/uw3XHJ/Ou00rwtZaR4l1rW7aW4a51pomuEkZSiGNNi7AACOOuSar6f4LstLtvEEVje38La/cS3M8ySqskEkihSYmCjbjAIzkg1C0jbpy2+en/BKerv5/hZnBeCfCWieC/j9q2meGrL7FZt4fimMXmvJ85nIJy5J7DvSfGS61jUvFvh7wtp+j3GtWN3BPdXOnQamth9s2YAVpT1VQSSo65B7VfT4F2Sao2pL468cC+eMQtcjV18xowchC3l5K55xnFdd4u8DaR40s7aLVftMNxZuZLS+tJjFcW7EYJRx0zgZzkcD0pvaN+l/1/wA/wFHSTa0v/kvzsedfDLwx4o8O+PQ1t4NuPCfhqa0dbmzfXUv4mmBBR1Gdyt1B6g+1VvF/giSTxpq+reOvCWpeL9Klk82xu9L1BxJYRhQPKFsHTOCCdy5z16nFeh+FfAH/AAi2qSXv/CWeKNZ3wmL7Pq+pfaIlyQdwXaMNxjPoTWfqXwj0++1i6vrXxN4q0qO7maaez07VmigkdjljtIJGSTkAj2xQ9WvT9f6/IFs/M6DwRe6VqHgnTLjw9fXF/pxh2wT3LFpWCkrhyecggg59KZ4c8F6d4X1jW9S0+e8km1u5FzcLcT71RueEGOBz3yegzgAVo6FoeneGtEttI0W2W1srVdsUaknHOSSTySSSSTySazvDngvTvC+sa3qWnz3kk2t3IubhbifeqNzwgxwOe+T0GcACqbXM2iUvdSOhoooIyCKko5jxLpGsXUF3LFq8bWW3ebCSAKrKBypkU7sHB/OqGsaoZYPDGqW9i0ivITHbKQPnZMKuTx17+1ao8HW0jf6bqmrXsJOWt7i7LRt9QAK1L/SLLUtN+wXMIEAA2BPlMeOhX0xU2dvuPFnhK1Vzklytrq+bVO63vZeS0120OS1bTr0X+j3ur3Pm6jNqEQit4ifLgTqwA79BljXdVi6b4XtNOvheSXN5fXKLtjlvZvMMYPULwMVtVS0VjqweHlScpzVnK3W706t9/wAlb0Ciiig9AKKKKACiiigDF/5j9p/1wl/9Cjq/qWradotmbvWL+10+2BCma6mWJAT0G5iBVD/mP2n/AFwl/wDQo68u8UX3hi6+L2uXvxDMFzpnhyytoNPsLiPzFlmnDOxWI/6xztAxgjp6ZC62/r+ug0tGz2Kw1Gy1WyS80u8t721k+5PbSrIjfRlJBqzXivwrkvtM+LWs2B0BfDel6xpqapbaQrH9xtk8reyj5Y3bklRjACjtXtVU1on3/wCGJ6tf13AkAEk4A6k1T07V9N1iKWXSNQtb+OGQxSPazrKEcdVJUnBGRx15qj4p8J6Z4x0yLTtcE8lmkyyvDFO0azYBG19pBK85x7CuH+BttBZWPjG1tIUgt4fE93HFFGuFRQEAAHYAUo6tp9r/AIr/ADG9En/XX/I9SooqtqNkmpaZc2Msk0SXMTRNJBIUdQwxlWHIPPBpPbQEMTWNMl1d9Kj1G0fUY4/NezWdTMqZxuKZyBkjnHerleOeGPC+keEP2jP7M0C0FtbDwuXb5izSObkZdmPLMfU17HVfZT/rdr9A6tf1smFU77WNM0ya2h1LUbSzlupBFbpcTrG0znoqAn5j7CrleJ/EzwTouheIfC+t2cEkmp6l4utGnu7iVpX2lmPlqT91B2A9B1wKS1kl3aX3uwdG/X8Fc9sooooAr31/Z6ZZyXepXUFpbRDMk1xII0QepY8Cn21zBe2sVzZzR3FvMgeKWJwyOpGQwI4II7ivPvi34J0XV/DGteIdUgkuryw0idbWOWVjDCwRj5gj6b+evsPSug+G3/JLfC//AGCbb/0UtEdU/K343/yB6W87/hb/ADOmooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDF/5j9p/1wl/9Cjrh/F3g/WNL+KEHj3wv4csfEk72otp7G4uFglidfuzRSP8AKDt+U8ZwOOpx3H/MftP+uEv/AKFHW1R1TDo0cF4I8PeIJvFWpeM/Gtvb2OpXtulna6dby+aLO3UlsM44ZyxySOOPfA72iin5IOtzG8UalrelaQLjw1oH9v3nmBTafbUtcKc5be4I4449684+F8Pj7w9rOpWus+BPs1jrWsT6hNe/2xA/2QSAfLsXJfBUDIx16V7BRSjo7/10/wAgeqt/X9ahUN7LPBYTy2dv9quI42aKDeE81gOF3HgZPGT0qaik9gR4gJficPikfGX/AArHrpX9m/Y/7ftv+em/fv8A0xj8a9shZ3gjaaPypGUFo92dpxyM98U+iq6W/rv+obu4V458SU+IXiPWdMg0vwB5tnomtRX8N3/bNuv2xYicDY2Cm7PfOPQ17HRS6p9h9GjK8N6hq+p6LHc+IdE/sO+ZmD2X2tLnaAeDvQYORz7Vq0UUCOB+KM/i660W90Hwv4Q/tm31Kwkhkvf7Tit/s7OCuNj8tgYPUelSfC6XxXbeHbTRPFXhT+xI9LsobeG5/tGK4+1FV2k7U+50B5J613VFEdL+f/B/zB62/rt/kFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAYv/ADH7T/rhL/6FHW1WL/zH7T/rhL/6FHW1QAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAYv/MftP8ArhL/AOhR1tVi/wDMftP+uEv/AKFHW1QAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAf/Z)

Figure 4

## Conclusion

1. The final distribution of pennies does not depend on the order in which the machine processes the coin pairs as it divides the number in the first box by 2 and puts the remainder in the same box then puts the result in the next box. These steps are repeated till each box has a 0 or 1.
2. The minimum number of boxes needed to distribute n pennies is the minimum number of bits needed to represent a decimal number which is **ceil(Log2(n+1))** as n is the number of pennies.
3. Using the greedy algorithm, the machine needs **ceil(Log2(n+1)) – 1** iterations to distribute the pennies before stopping.

# Task 6

## Assumptions

• The user will input a positive integer number from 0 to 63.

• The source peg is A, the destination peg is D, and the other two auxiliary pegs are B and C.

## Problem Description

There are eight disks of different sizes and four pegs Initially, all the disks are on the first peg in order of size, the largest on the bottom and the smallest on the top. Use dynamic programming method to transfer all the disks to another peg by a sequence of moves. Only one disk can be moved at a time, and it is forbidden to place a larger disk on top of a smaller one. Does the dynamic programming method can solve the puzzle in 33 moves? If not then design an algorithm that solves the puzzle in 33 moves.

## Detailed Solution

Since this problem is a deviation from the Towers of Hanoi, it can be solved using the same recursive concept, but with the aid of dynamic programming to save our answers and greatly reduce the recursive calls made.

The trivial cases are moving 0, 1, or 2 disks from A to D and the answer for these cases are 0, 1, and 3 respectively.

In the case of moving n disks, where n > 2 and n < 64, k disks should be moved from A to one of the 2 auxiliary pegs, B or C, using the 4 pegs in order to achieve this. Then, the remaining disks are moved to the destination peg D using the remaining 3 pegs, which is exactly the Towers of Hanoi problem. Finally, the k disks are moved to D using all four pegs.

So, we have to identify which value to k would result in moving all disks from A to D using the minimum number of steps.

Remember that: The number of steps to move n disks from the source to the destination peg using only 3 pegs = (Towers of Hanoi relation). Note that moving more than 63 disks from A to D results in an overflow because of this relation.

Thus, the recurrence relation for our problem is:

Dynamic programming can be used to store the values calculated by this recursive function to greatly reduce the number of recursive calls, such that reveDp[i].first represents the minimum number of steps to move i disks from A to D, and reveDp[i].second represents the number of disks to be moved using the 4 pegs (k).

After obtaining the minimum number of steps and the k value, messages are printed recursively to show the exact steps of moving the disks. Note that printing the messages cannot be achieved in any other way than recursively as each message specifies a certain disk number to be moved, a certain source peg, and a certain destination which is unique for this case and cannot be known using any previously stored values.

Pseudo-code

int reveDP[n];

/\*The dp array used to store the minimum number of steps to move the disks from souce to destination (s)

\*and the number of disks to be moved using all 4 pegs (k)

\*/

ALGORITHM getRevePair(n)

//INPUT: n - number of disks to be moved.

/\*OUTPUT: minimum number of steps to move the disks from souce to destination (s)

\* and the number of disks to be moved using all 4 pegs (k).

\*/

int s < -infinity;

int k < --1; //a flag to indicate a base case

if n = 0

return s < -0 and k < --1;

if n = 1

return s < -1 and k < --1;

if n = 2

return s < -3 and k < --1;

for i = 1 to n - 1 do

int temp < -(2 \* getRevePair(i).first + pow(2, n - i) - 1);

if temp < s

s = temp;

k = i;

return s and k;

ALGORITHM hanoiTransfer(n, from, to, aux, offset)

/\*INPUT: n - number of disks to be moved using 3 pegs.

\* from: the name of the souce peg.

\* to: the name of the destination peg.

\* aux: the name of the third peg

\* offset: an offset to be added to the number of disk to be moved

as some of the disks are already moved using the four pegs before calling this algorithm

\*/

//OUTPUT: prints the steps of moving the disks from the souce to the destination using three pegs.

if n = 0

return;

// move all the disks above this disk from the source to the aux peg

hanoiTransfer(n - 1, from, aux, to, offset);

print the message of moving disk n + offset from the source to the destination;

// move the previously moved disks from the aux to the destination peg

hanoiTransfer(n - 1, aux, to, from, offset);

Algorithm reveTransfer(n, from, to, aux1, aux2)

/\*INPUT: n - number of disks to be moved using 4 pegs.

\* from: the name of the source peg.

\* to: the name of the destination peg.

\* aux1: the name of the third peg

\* aux2: the name of the fourth peg

\*/

//OUTPUT: prints the steps of moving the disks from the source to the destination using four pegs.

if n = 0

return;

if n = 1

move the only disk from source to destination.

if n = 2

move the top disk from source to aux

move the bottom disk from source to destination

move the first disk from aux to destination

//transition:

// transfer k disks from the source to aux2

reveTransfer(reveDp[n]->k, from, aux2, aux1, to);

//transfer the remaining disks from souce to destination using all pegs except aux2

hanoiTransfer(n - reveDp[n]->k, from, to, aux1, reveDp[n].second);

//move the previously transferred disks from aux2 to the destination

reveTransfer(reveDp[n] ->, aux2, to, aux1, from);

For simplicity, the algorithm is divided into three sub algorithms that interact with each other.

## Complexity Analysis

* Time complexity for hanoiTransfer = , since the number of recursion calls per function call = 2, the number of disks = n, and is decreased by 1 at every recursive level.
* Time complexity for getRevePair = , since getRevePair (n) calls getRevePair (1) : getRevePair (n - 1) and so on, and each one of these calls gets evaluated in O(1) using dynamic programming.

## Comparison With Another Algorithm

An alternative solution to moving the disks from the source to the destination pegs is using the exact same approach as the Towers of Hanoi with slight modifications

• Using two auxiliary pegs instead of one.

• Instead of moving all the disks above the bottom one (with number n) to the auxiliary peg, then moving the bottom disk to the destination, and finally moving the previously transferred disks from the auxiliary peg to the destination, we use a different approach. All of the disks above the disk number n – 1 (the one above the bottom disk) are moved to one of the auxiliary pegs, then the two bottom disks are moved from source to destination with the help of the second auxiliary peg, and finally the previously transferred disks are moved again from the first auxiliary peg to the destination.

This algorithm uses pure recursion. Thus, it is less efficient than the dp solution, it also does not guarantee the minimum number of steps to transfer the disks from source to destination, but it is much more straight forward to implement.

## Sample Output

• Output in case of zero disks (base case).

Text

Description automatically generated

Figure 1

• Output in case of one disk (base case).

Text

Description automatically generated

Figure 2

• Output in case of 2 disks (base case).

Figure 3

Text

Description automatically generated

• Output in case of 5 disks (randomly selected number).

Text

Description automatically generated

Figure 4

• Output in case of 8 disks (the description given number).

A picture containing graphical user interface

Description automatically generated

Figure 5

Graphical user interface, text

Description automatically generated

Figure 6

## Conclusion

• The Reve puzzle can be solved using dynamic programming recursively by moving a subset of disks to an auxiliary peg using 4 pegs. Then, moving the remaining disks to the destination using the Towers of Hanoi algorithm. Finally, retransferring the previously moved pegs from the auxiliary peg the destination.

• The puzzle can be solved using pure recursion with easier implementation, but the optimum solution and the dp algorithm efficiency are to be sacrificed.