Let $B = 64$, and the actual block alphabet $\Sigma = \{0, \ldots, 9, a, \ldots, z, A, \ldots, Z, ., "\text{ (period)}, "\text{ (space)}\}$. To translate to more usual $\{0, \ldots, 63\} = [64]$, let us encode the digits $0 \ldots 9$ as $0 \ldots 9$, letters $a \ldots z$ as $10 \ldots 35$, capital letters $A \ldots Z$ as $36 \ldots 61$, period "." as 62 and space " " as 63. As usual, $\text{EOF}$ is encoded as 64.

(a) (3 points) With the convention above, characters in which position of the input string does the 3rd character of the SOLE encoding depend on? Assume that the input is a word “SoLe.”. Find the 3rd character of the SOLE encoding of this string. Explain how exactly you determined that character.

(Hint: One option is to do the full encoding. However, if you are smart, you do not need to!)

Solution: ***************** INSERT YOUR SOLUTION HERE ***************** 

(b) (3 points) Assume somebody sent you the SOLE encoding which reads “ON7DMa1”. What is the 3rd character of the original message? Justify your answer.

(Hint: One option is to do the full decoding. However, if you are smart, you do not need to!)

Solution: ***************** INSERT YOUR SOLUTION HERE ***************** 

(c) (3 points) Change the encoded string in part (b) in a minimum number of places, such that your answer in part (b) changes to $\text{EOF}$.

(Hint: One option is to do the full decoding, change letter, and then do full re-encoding. However, if you are smart, you do not need to!)

Solution: ***************** INSERT YOUR SOLUTION HERE ***************** 

(d) (Extra credit: 6 points)

Get full encoding in part (a) and full decodings in parts (b) and (c).\(^1\)

Solution: ***************** INSERT YOUR SOLUTION HERE ***************** 

\(^1\)Compare the complexity with correct solutions to parts (a)-(c).
You own a zoo with $n$ animals. Some pairs of animals cannot co-exist with each other (e.g., one will eat the other), while others can. You worked hard, and developed a complete set of $m$ pairs of animals which cannot co-exist. Design $O(m + n)$ algorithm to partition (if at all possible) the animals into two groups, such that each animal in a group can co-exist with every other animal in the same group.

(Hint: Use BFS on an appropriate graph and use the BFS tree to derive the only possible partition. Then verify that this partition is indeed OK.)

Solution: ******************* INSERT YOUR SOLUTION HERE *******************
A fellow Moe and his buddy Joe live in a city $G = (V, E)$ which is an undirected graph on $n$ vertices and $m$ edges, given in the adjacency list form. Moe lives in a vertex $a$ and owns a crazy dog Mimi, while Joe lives at a vertex $b$ and owns a crazy dog Kiki. This Sunday Moe wants to take Mimi to a veterinarian clinic located at vertex $c$, while Joe wants to take Kiki to the dance competition located at a vertex $d$. One problem, though: the dogs hate each other, and if one of them smells the other, all hell breaks lose. Luckily, they can smell each other only if within distance at most 15 in $G$, and you know that $dist_G(a, b), dist_G(c, d) > 15$. Moe and Joe would like to start at the same time (with Moe and Mimi at $a$ and Joe and Kiki at $b$) and get both dogs to their respective destinations $c$ and $d$ in the smallest number of steps $t$. A step consists of both dogs going to their respective neighboring vertices, or one dog going to a neighboring vertex, and the other dog staying put, barking at the pedestrians. Of course, such a step is possible only if the dogs stay within distance 16 or more both before and after the step.

Your job is to design an algorithm for Moe and Joe to compute $t$ (and the optimal route), if the route exists, and analyze its running time.

(a) Using one or more runs of the BFS algorithm on $G$, fill in the matrix $OK[x, y]$, where $OK[x, y] = 1$, if it is OK for Mimi to be at vertex $x$ and Kiki to be at vertex $y$ at the same time, and $OK[x, y] = 0$ otherwise. How long did it take you to fill this matrix $OK[x, y]$?

Solution: ****************** INSERT YOUR SOLUTION HERE ******************

(b) Design a graph $H = (V', E')$ whose vertex set consists of possible “ok configurations” for Mimi and Kiki, and whose edge set represents the possible single steps of your algorithm. Be sure to formally define $V'$ and $E'$ as functions of $V$ and $E$ and the matrix $OK$ from part (a). How long (in the worst case) did it take you to create an adjacency list for $H$ (not counting what you did in part (a))? What is the maximum $|V'|$ and $|E'|$?

Solution: ****************** INSERT YOUR SOLUTION HERE ******************

(c) Describe the original problem as a shortest path computation on $H$. Finally, solve the original problem, and help Moe and Joe. Analyze the overall running time of your algorithm, as a function of $n$ and $m$.

Solution: ****************** INSERT YOUR SOLUTION HERE ******************