Q1 Team Name
0 Points

Pasta_Sandwich

Q2 Commands
10 Points

List the commands used in the game to reach the ciphertext.

go, back, read

Q3 CryptoSystem
10 Points

What cryptosystem was used in this level?

Playfair cipher and Morse code

Q4 Analysis
20 Points

What tools and observations were used to figure out the cryptosystem? (Explain in less than 300 words)

i) Starting from Level 2, we entered "go" as the first command. A pattern of dots and dashes was visible next. The patterns somewhat looked like morse code, so we used https://www.electronics-notes.com/articles/ham_radio/morse_code/characters-table-chart.php for the English translation.
ii) We used the mapping between morse code and English alphabets to write a C++ code that decodes the morse code. The decoded word was CRYPTANALYSIS.
iii) The spirit in the cave said "You have been blessed, my child. Keep in mind that you must always believe in yourself and PLAY FAIR." This hinted that the cipher system used might be a Playfair cipher.
iv) To reach the ciphertext, we used "back" and "read" commands. Considering the cipher system to be PLAYFAIR, CRYPTANALYSIS seems to be a valid keyword used for encryption, as it does not contain the letter 'j'.
v) We created the Key Square using the assumed key and used it for manual decryption.
vii) There were a few words with extra 'x' in the deciphered text. These 'x' were used so that both letters of the bigrams are not the same. For example, The decrypted text contained words like 'WILXL NEXED'. The Xs were removed from such words.
viii) In the Playfair cipher, The 'J's are replaced with 'I's as the key square doesn't contain any 'J'. There was a word with the spelling 'IOY' which has no meaning, so we replaced the 'I' with 'J' and the new word formed was 'JOY' which is meaningful.
ix) We also note that since PLAYFAIR Cipher is used, symbols like ':' , '_' , '"' were not encrypted.
x) The spaces were kept unaltered in the decryption process.
xi) Given the text decrypted using the PLAYFAIR cipher system was meaningful we were confident that the cipher system used was PLAYFAIR.

Q5 Decryption Algorithm
15 Points

Briefly describe the decryption algorithm used. Also mention the plaintext you deciphered. (Use less than 350 words)

i) We created a 5X5 Key square using the keyword 'CRYPTANALYSIS':
The initial alphabets of the Key Square are the unique letters from the keyword in their order of appearance in the keyword (Thus in our Key Square, S letter is only taken once).
The remaining boxes are filled with letters from the alphabet in alphabetical order.
Key Square:
C, R, Y, P, T,
A, N, L, S, I,
B, D, E, F, G,
H, K, M, O, Q,
U, V, W, X, Z
ii) We used the Key Square to decrypt the ciphertext. Rules of decryption are as follows:
We take each of the bigrams from the ciphertext and ignore the whitespaces and the special characters. Then we look for the bigrams in the Key Square. The letters are replaced using the below-mentioned rules:
-If both the letters are in the same column: Take the letter above each one, wrap around if required.

Assignment 2

GRADED

GROUP
Manjyot Singh Nanra
Ayush Sahni
Sharanya Saha
View or edit group

TOTAL POINTS
65 / 65 pts

QUESTION 1
Team Name — **0** / 0 pts

QUESTION 2
Commands — **10** / 10 pts

QUESTION 3
CryptoSystem — **10** / 10 pts

QUESTION 4
Analysis — **20** / 20 pts

QUESTION 5
Decryption Algorithm — R **15** / 15 pts

QUESTION 6
Password — **10** / 10 pts

QUESTION 7
Code — **0** / 0 pts

-If both the letters are in the same row: Take the letter to the left of each one, wrap around if required.
-If neither of the above rules is true: Form a rectangle with the two letters and take the letters on the horizontal opposite corner of the rectangle.
iii) After following the above rules, the decrypted text received was 'BE WARY OF THE NEXT CHAMBER, THERE IS VERY LITTLE IOY THERE. SPEAK OUTX THE PASSWORD "ABRA_CA_DABRA" TO GO THROUGH. MAY YOU HAVE THE STRENGTH FOR THE NEXT CHAMBER. TO FIND THE EXIT YOU FIRST WILXL NEXED TO UTTER MAGIC WORDS THERE'
The text contained a few extra 'X's which were removed to make the words meaningful. We decrypted the ciphertext manually.
iv) The password received from the decrypted text was- 'ABRA_CA_DABRA' (Note that the underscore were not encrypted)

Q6 Password

10 Points

What was the final command used to clear this level?

ABRA_CA_DABRA

Q7 Code

0 Points

Upload any code that you have used to solve this level

assignment2.cpp Download

```cpp
#include<bits/stdc++.h>
using namespace std;
string decode_morse(string morse)
{

        if(morse == ".-")
        {return "a";}
        else if(morse == "-...")
        {return "b";}
        else if(morse == "-.-.")
        {return "c";}
        else if(morse == "-..")
        {return "d";}
        else if(morse == ".")
        {return "e";}
        else if(morse == "..-.")
        {return "f";}
        else if(morse == "--.")
        {return "g";}
        else if(morse == "....")
        {return "h";}
        else if(morse == "..")
        {return "i";}
        else if(morse == ".---")
        {return "j";}
        else if(morse == "-.-")
        {return "k";}
        else if(morse == ".-..")
        {return "l";}
        else if(morse == "--")
        {return "m";}
        else if(morse == "-.")
        {return "n";}
        else if(morse == "---")
        {return "o";}
        else if(morse == ".--.")
        {return "p";}
        else if(morse == "--.-")
        {return "q";}
        else if(morse == ".-.")
        {return "r";}
        else if(morse == "...")
        {return "s";}
        else if(morse == "-")
        {return "t";}
        else if(morse == "..-")
        {return "u";}
        else if(morse == "...-")
        {return "v";}
        else if(morse == ".--")
        {return "w";}
        else if(morse == "-..-")
        {return "x";}
        else if(morse == "-.--")
        {return "y";}
        else if(morse == "--..")
        {return "z";}
        return "";
}
int main() {
        string morse_code="-.-. .-. .-.-- .--. - .- -. .- .-.. -.-- ... .. ...";
        string decoded="";
        string ch="";
        for(int i=0;i<morse_code.length();i++)
```

```
        for(int i=0;i<morse_code.length();i++)
        {
            if(morse_code[i]!=' ')
                ch=ch+morse_code[i];
            else
            {
                decoded=decoded+decode_morse(ch);
                ch="";
            }
        }
        decoded=decoded+decode_morse(ch);
        cout<<decoded;
        return 0;
}

```