**LAB 06**

**ENCRYPT:**

var chars = "abcdefghijklmnopqrstuvwxyz";

function Encrypt(plaintext, key ) {

var klen = key.length;

var colLength = plaintext.length / klen;

var ciphertext = "";

k = 0;

for (i = 0; i < klen; i++) {

while (k < 26) {

t = key.indexOf(chars.charAt(k));

arrkw = key.split("");

arrkw[t] = "\_";

key = arrkw.join("");

if (t >= 0) break;

else k++;

}

for (j = 0; j < colLength; j++) {

ciphertext += plaintext.charAt(j \* klen + t);

}

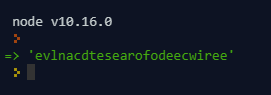
}

return ciphertext;

}

Encrypt("wearediscoveredfleeatonce","zebras");

**OUTPUT:**



**DECRYPT:**

var chars = "abcdefghijklmnopqrstuvwxyz";

function Decrypt(ciphertext, keyword) {

var klen = keyword.length;

// first we put the text into columns based on keyword length

var cols = new Array(klen);

var colLength = ciphertext.length / klen;

for (i = 0; i < klen; i++) cols[i] = ciphertext.substr(i \* colLength, colLength);

// now we rearrange the columns so that they are in their unscrambled state

var newcols = new Array(klen);

j = 0;

i = 0;

while (j < klen) {

t = keyword.indexOf(chars.charAt(i));

if (t >= 0) {

newcols[t] = cols[j++];

arrkw = keyword.split("");

arrkw[t] = "\_";

keyword = arrkw.join("");

} else i++;

}

// now read off the columns row-wise

var plaintext = "";

for (i = 0; i < colLength; i++) {

for (j = 0; j < klen; j++) {

plaintext += newcols[j].charAt(i);

}

}

return plaintext;

}

Decrypt("evlnacdtesearofodeecwiree","zebras");

**OUTPUT:**

