

Name - Panwar Karan  
U. Roll No - 1121097  
BCA 6 (B)

PBC-601

4

3

```
def generateKey(string, key):
```

```
    key = list(key)
```

```
    if len(string) == len(key):
```

```
        return key
```

```
    else:
```

```
        for i in range(len(string) - len(key)):
```

```
            key.append(key[i % len(key)])
```

```
    return "".join(key)
```

```
def cipherText(string, key):
```

```
    cipher_text = []
```

```
    for i in range(len(string)):
```

```
        x = (ord(string[i]) + ord(key[i])) % 26
```

```
        x += ord('A')
```

```
        cipher_text.append(chr(x))
```

```
    return "".join(cipher_text)
```

```
def originalText(cipher_text, key):
```

```
    orig_text = []
```

```
    for i in range(len(cipher_text)):
```

```
        x = (ord(cipher_text[i]) -
```

```
            ord(key[i]) + 26) % 26
```

```
        x += ord('A')
```

```
        orig_text.append(chr(x))
```

```
    return "".join(orig_text)
```



i) `__name__ == "__main__":`

`String = "`

`Keyword =`

`key = generatekey(String, keyword)`

`ciphertext = cipherText(String, key)`

`print("Ciphertext : " + ciphertext)`

`print("Original/Decrypted text : ", originalText(ciphertext, key))`



Name - Parag Ramani  
U. Roll no - 1121099  
BCA - 6 C B

P.B.C - 602

5 -

```
def encrypt(text, s):  
    result = ""
```

```
    for i in range(len(text)):  
        char = text[i]
```

```
        if (char.isupper()):
```

```
            result += chr((ord(char) + s - 65) % 26 + 65)
```

```
        else:
```

```
            result += chr((ord(char) + s - 97) % 26 + 97)
```

```
    return result
```

```
text = "Attack Attack from North"
```

```
s = 4
```

```
print "Text : " + text
```

```
print "Shift : " + str(s)
```

```
print "Cipher" + encrypt(text, s)
```



(4)

Name - Paras Raut  
U. Roll no - 1121097  
BCA - 6 (B)

PBL-601

```
import math, random
```

```
def generateOTP():
```

```
    digits = "0123456789"  
    OTP = ""
```

```
    for i in range(4):
```

```
        OTP += digits[math.floor(random * random() * 10)]
```

```
    return OTP
```

```
if __name__ == "__main__":
```

```
    print("OTP of 4 digits:", generateOTP())
```