```
import string

def caesar_decrypt(ciphertext, shift):
    alphabet = string.ascii_uppercase
    shifted_alphabet = alphabet[shift:] + alphabet[:shift]
    table = str.maketrans(alphabet, shifted_alphabet)
    return ciphertext.translate(table)

def polyalphabetic_decrypt(ciphertext, keyword):
    alphabet = string.ascii_uppercase
    decrypted_text = ''
    keyword_repeated = (keyword * (len(ciphertext) // len(keyword))) + keyword[:len(ciphertext) %
len(keyword)]

    for char, keyword_char in zip(ciphertext, keyword_repeated):
        if keyword_char in alphabet:
            shift = string.ascii_uppercase.index(keyword_char)
            decrypted_text += caesar_decrypt(char, shift)
        else:
            decrypted_text += char

    return decrypted_text

def substitution_decrypt(ciphertext, substitution_dict):
    decrypted_text = ''.join(substitution_dict.get(char, char) for char in ciphertext)
    return decrypted_text

def monoalphabetic_decrypt(ciphertext, key):
    substitution_dict = dict(zip(key, string.ascii_uppercase))
```

```
    return substitution_decrypt(ciphertext, substitution_dict)


def vigenere_decrypt(ciphertext, keyword):
    decrypted_text = ''
    keyword_repeated = (keyword * (len(ciphertext) // len(keyword))) + keyword[:len(ciphertext) % len(keyword)]

    for char, keyword_char in zip(ciphertext, keyword_repeated):
        if char.isalpha() and keyword_char.isalpha():
            try:
                shift = string.ascii_uppercase.index(keyword_char) - string.ascii_uppercase.index('A')
                decrypted_text += caesar_decrypt(char, shift)
            except ValueError:
                decrypted_text += char
        else:
            decrypted_text += char

    return decrypted_text


def main():
    encrypted_text = input("Enter the encrypted text: ").upper()

    print("Choose the cipher type:")
    print("1. Caesar")
    print("2. Polyalphabetic")
    print("3. Substitution")
    print("4. Monoalphabetic")
    print("5. Vigenere")
    cipher_type = int(input("Enter the number corresponding to the cipher type: "))
```

```
    if cipher_type == 1:
        shift_caesar = int(input("Enter the Caesar shift: "))
        result = caesar_decrypt(encrypted_text, shift_caesar)
    elif cipher_type == 2:
        keyword_poly = input("Enter the Polyalphabetic keyword: ").upper()
        result = polyalphabetic_decrypt(encrypted_text, keyword_poly)
    elif cipher_type == 3:
        substitution_dict = dict(input("Enter the Substitution parameters (e.g., {'B': 'A', 'U': 'V'}): "))
        result = substitution_decrypt(encrypted_text, substitution_dict)
    elif cipher_type == 4:
        key_mono = input("Enter the Monoalphabetic key: ").upper()
        result = monoalphabetic_decrypt(encrypted_text, key_mono)
    elif cipher_type == 5:
        keyword_vigenere = input("Enter the Vigenere keyword: ").upper()
        result = vigenere_decrypt(encrypted_text, keyword_vigenere)
    else:
        print("Invalid cipher type selected.")
        return

    print("Decrypted Result:", result)

if __name__ == "__main__":
    main()
```