

```
"""
```

```
Created on Mon Mar 9 20:54:24 2020
```

```
@author: mahika banerjee
```

```
"""
```

```
def hex2dec(hx_str):  
    decval = int(hx_str,16)  
    binval = bin(decval)[2:]  
    binval = (4 - len(binval))*'0'+ binval  
    return binval
```

```
while True:
```

```
    input_hex = input('Enter your hexadecimal value here, type "0" to exit: ')
```

```
    if input_hex == '0':
```

```
        print('Thank you!')
```

```
        break
```

```
    elif len(input_hex) != 2:
```

```
        print('Enter 2 digits only')
```

```
        continue
```

```
    else:
```

```
        try:
```

```
            binary_one = list(hex2dec(input_hex[0]))
```

```
            binary_two = list(hex2dec(input_hex[1]))
```

```
        except:
```

```
            print('Enter only Hexadecimal numbers')
```

```
            continue
```

```
    dbg = 0
```

```
    if dbg == 1:
```

```
        print('Input in table form:')
```

```
        print("    ['A', 'A', '~A', '~A']")
```

```
        print(' B ', binary_one)
```

```
        print('~B ', binary_two)
```

```
        print("    ['~C', 'C', 'C', '~C']")
```

```
    #Condition #0
```

```
    if dbg == 1:
```

```
        print('Cond #0')
```

```
    if binary_one + binary_two == list('11111111'):
```

```
        output = '1 + '
```

```
    elif binary_one + binary_two == list('00000000'):
```

```
        output = '0 + '
```

```
    else:
```

```
        output = ''
```

```
        #Check Condition #1
```

```
        if dbg == 1:
```

```
            print('Cond #1, Output:', output)
```

```
        if binary_one == list('1111'):
```

```
            output += 'B + '
```

```
            binary_one = list('0000')
```

```

if binary_two == list('1111'):
    output += '~B + '
    binary_two = list('0000')

#Check Condition #2
if dbg == 1:
    print('Cond #2, Output:', output)
if binary_one[0:2]+binary_two[0:2] == list('1111'):
    output += 'A + '
    binary_one[0:2] = ['0','0']
    binary_two[0:2] = ['0','0']
if binary_one[1:3]+binary_two[1:3] == list('1111'):
    output += 'C + '
    binary_one[1:3] = ['0','0']
    binary_two[1:3] = ['0','0']
if binary_one[2:4]+binary_two[2:4] == list('1111'):
    output += '~A + '
    binary_one[2:4] = ['0','0']
    binary_two[2:4] = ['0','0']

#Check Condition #3
if dbg == 1:
    print('Cond #3, Output:', output)
if binary_one[0]+binary_one[3]+binary_two[0]+binary_two[3] == '1111':
    output += '~C + '
    binary_one[0] = '0'
    binary_one[3] = '0'
    binary_two[0] = '0'
    binary_two[3] = '0'

#Check Condition #4
if dbg == 1:
    print('Cond #4.1, Output:', output)
if binary_one[0:2] == list('11'):
    output += 'AB + '
    binary_one[0:2] = ['0','0']
if binary_one[1:3] == list('11'):
    output += 'BC + '
    binary_one[1:3] = ['0','0']
if binary_one[2:4] == list('11'):
    output += '~AB + '
    binary_one[2:4] = ['0','0']

if dbg == 1:
    print('Cond #4.2, Output:', output)
if binary_two[0:2] == list('11'):
    output += 'A~B + '
    binary_two[0:2] = ['0','0']
if binary_two[1:3] == list('11'):
    output += '~BC + '

```

```

    binary_two[1:3] = ['0','0']
if binary_two[2:4] == list('11'):
    output += '~A~B + '
    binary_two[2:4] = ['0','0']

#Check Condition #5
if dbg == 1:
    print('Cond #5.1, Output:', output)
if binary_one[0]+binary_two[0] == '11':
    output += 'A~C'
    binary_one[0] = '0'
    binary_two[0] = '0'

if dbg == 1:
    print('Cond #5.2, Output:', output)
if binary_one[1]+binary_two[1] == '11':
    output += 'AC + '
    binary_one[1] = '0'
    binary_two[1] = '0'

if dbg == 1:
    print('Cond #5.3, Output:', output)
if binary_one[2]+binary_two[2] == '11':
    output += '~AC + '
    binary_one[2] = '0'
    binary_two[2] = '0'

if dbg == 1:
    print('Cond #5.4, Output:', output)
if binary_one[3]+binary_two[3] == '11':
    output += '~A~C + '
    binary_one[3] = '0'
    binary_two[3] = '0'

#Condition #6
if dbg == 1:
    print('Cond #6.1, Output:', output)
if binary_one[0]+binary_one[3] == '11':
    output += 'B~C + '
    binary_one[0] = '0'
    binary_one[3] = '0'

if dbg == 1:
    print('Cond #6.2, Output:', output)
if binary_two[0]+binary_two[3] == '11':
    output += '~B~C + '
    binary_two[0] = '0'
    binary_two[3] = '0'

#Condition #7

```

```

if dbg == 1:
    print('Cond #7.1, Output:', output)
if binary_one[0] == '1':
    output += 'AB~C + '
    binary_one[0] = '0'

if dbg == 1:
    print('Cond #7.2, Output:', output)
if binary_one[1] == '1':
    output += 'ABC + '
    binary_one[1] = '0'

if dbg == 1:
    print('Cond #7.3, Output:', output)
if binary_one[2] == '1':
    output += '~ABC + '
    binary_one[2] = '0'

if dbg == 1:
    print('Cond #7.4, Output:', output)
if binary_one[3] == '1':
    output += '~AB~C + '
    binary_one[3] = '0'

if dbg == 1:
    print('Cond #7.5, Output:', output)
if binary_two[0] == '1':
    output += 'A~B~C + '
    binary_two[0] = '0'

if dbg == 1:
    print('Cond #7.6, Output:', output)
if binary_two[1] == '1':
    output += 'A~BC + '
    binary_two[1] = '0'

if dbg == 1:
    print('Cond #7.7, Output:', output)
if binary_two[2] == '1':
    output += '~A~BC + '
    binary_two[2] = '0'

if dbg == 1:
    print('Cond #7.8, Output:', output)
if binary_two[3] == '1':
    output += '~A~B~C + '
    binary_two[3] = '0'

print(output[:-3])

```