1. Problem Statement:

Overall, the object of assignment 1 is very lucid.

a) **For exercise 1**, we need to write a bool function which compares the identity of two vectors while ignoring the order and multiplicities.

b) **For exercise 2**, it is required that we develop a pointer function for searching a named character sequence and return the pointer which points the first elements; if not found, return a nullptr.

2. Analysis:

a) Exercise 1:

Input: the sizes and ALL elements of 2 vectors

Output: display/ print the identity of the two vector

b) Exercise2:

Input: a source character sequence to be checked and a obj character sequence to be checked out from the source;

Output: display the pointer pointing to the first found character; if not found, return a NULL pointer.

3. Design the list of steps (algorithms to solve the problem)

a) Exercise 1:

Functions:

same_vec(vector<int> a, vector<int> b): bool

Declare and INITIALIZE bool flag same_element to TRUE
LOOP.outer:
IF j < a.size() AND element_same == true
SET same_element =FALSE
ELSE
END LOOP.outer
LOOP.inner:
IF a[j] == b[i] AND i < a.size
SET same_element to TRUE
END LOOP
ELSE
i = i + 1,
REPEAT UNTIL i = a.size
THEN SET same_element to FALSE
LOOP.inner END
j = j +1
REPEAT LOOP.outer

RETURN same_element

isInt(string str): bool

IF str[0] = '-' && str.size() >= 2
SET isDigit to TRUE

```
    IF str.size()> 10
    SET isDigit to FALSE;

    ELSE IF str[0] >= '0' AND str[0] <= '9'
        SET isDigit to TRUE;
    IF (str.size()> 9)
        SET isDigit to FALSE;
    ELSE
        SET isDigit to false;
    INITIALIZE i to 1
    LOOP:
        IF i >= str.size()
            END LOOP
        IF str[i] < '0' OR str[i] > '9'
            SET isDigit to false;
        i= i + 1
    END LOOP
    RETURN isDigit
    main(void): int
        PRINT  “Enter size of the 1st vector:”
        STORE INPUT in STRING str

strToInt(string str): int

    DECLARE istringstream streami;
    DO streami.str(str)
    DECLARE i;
    INPUT streami to i;

    RETURN i;
main(): int
    PROMPT the user to INPUT the size of the vector
    CALL isInt(string str) and strToInt(string str) TO PARSE input string
    IF PARSE input String to int temp
        SET count1 to temp
    ELSE
        ASK repeatedly for correct int INPUT
    PROMPT the user to INPUT the ELEMENT of the vector once a time
    CALL isInt(string str) and strToInt(string str) TO PARSE input string
    IF PARSE input String to int temp
        APPEND temp to the first vector<int> vi1
    ELSE
        ASK repeatedly for correct int INPUT
    REPEAT the above for the second vector<int> vi2
```

```
PRINT vi1, vi2
IF same_vec(vi1, vi2) AND same_vec(vi2, vi1)
    PRINT "The 2 vectors are identical."
ELSE
    PRINT "The two vectors are different."
RETURN 0
```

b) Exercise 2: find characters one by one

findC(char const* source, char const* obj) char*

```
    INITIALIZE char *ptr to nullptr
    INITIALIZE int count to 0
    DECLARE int i, j , k
    IF strlen(source) >= strlen(obj)
    FOR (i = 0, j = 0; i < strlen(source) -strlen(obj)+1; i++)
        IF source[i] == obj[j])
            SET k to i
        FOR (count = 0; source[k] == obj[j] && j < strlen(obj); k++, j++)
            SET count to count + 1
                IF   count == strlen(obj)
                    SET ptr to (char*)&source[i];
                    SET i to strlen(source);
                ELSE
                     SET j to 0
    Return ptr;
```

strToCharArray(void): char*

```
    INTIALIZE string str to "";
    SET str to INPUT

    DECLARE istringstream streami
    DO streami.str(str);
    SET char *pc to new char[str.size()] ;
    INPUT streami to pc;

    return pc;
```

int main()

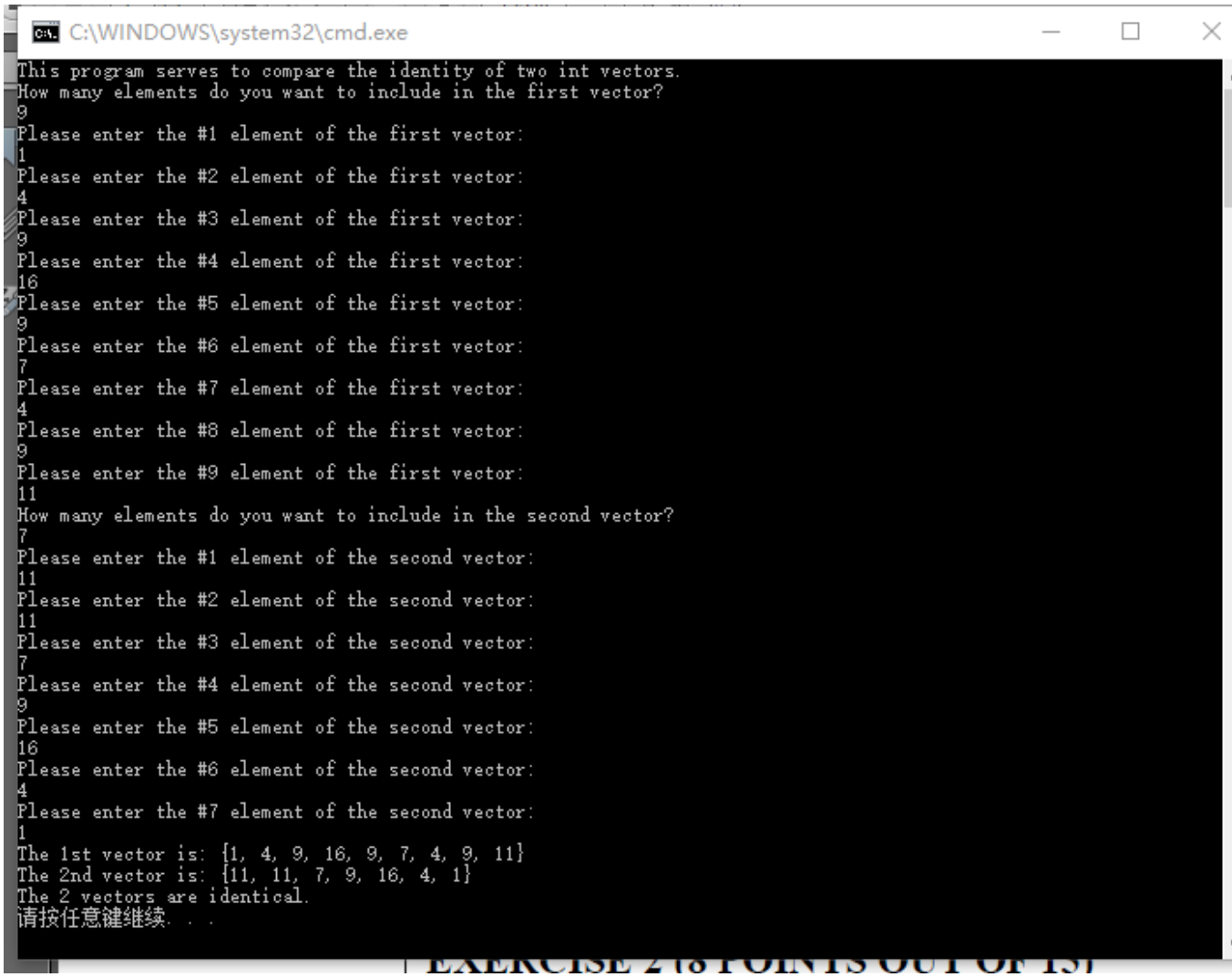
```
    DECLARE char const* source
    DECLARE char const* obj

    PROMPT the user to INPUT the source string
    SET source TO strToCharArray()

    PROMPT the user to INPUT the obj string
```

SET obj to strToCharArray()

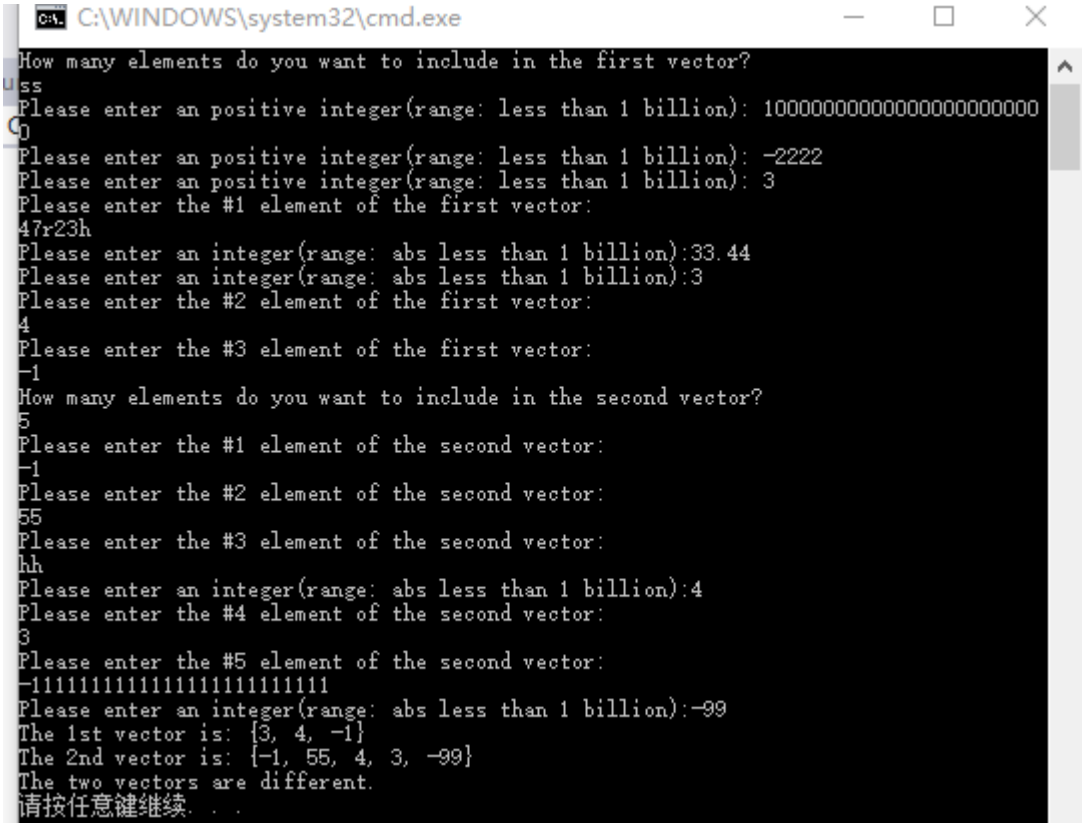
PRINT (int*)findC(source, obj)

return 0

4. Implementation: indicate the name of the file. 50%
a) Exercise 1: Exercise1_Assgnmet 1 _Zheng.Sun15_1507820
b) Exercise 2: Exercise2_Assgnmet 1 _Zheng.Sun15_1507820
5. Testing
a) Exercise 1:
Commanded Test in Example:



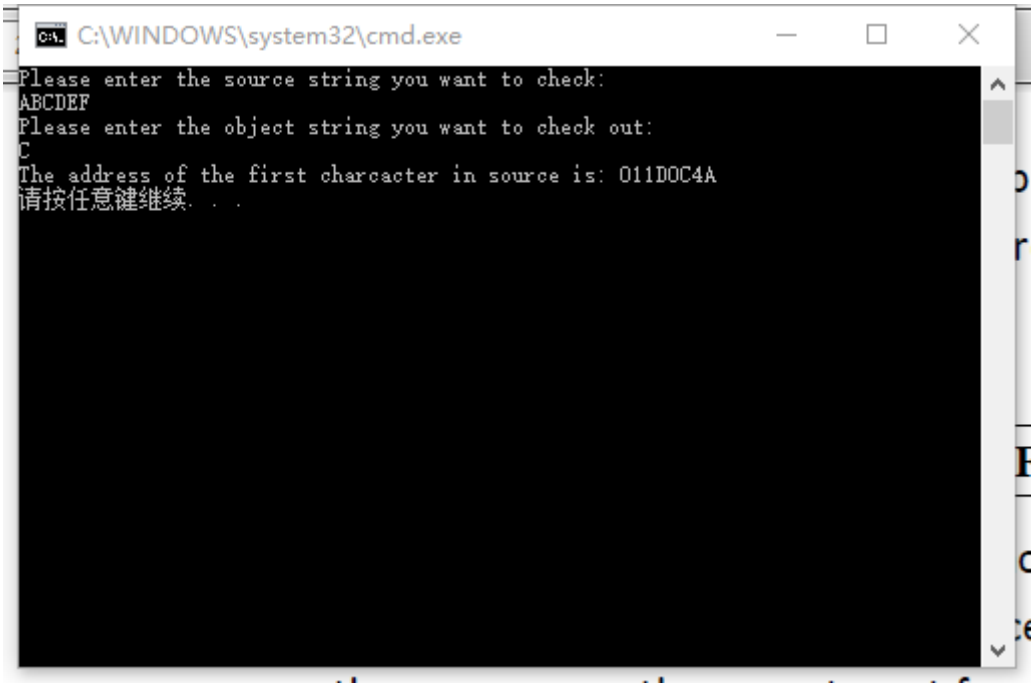
1. **Self- test: to check if erroneous input will affect the result.**

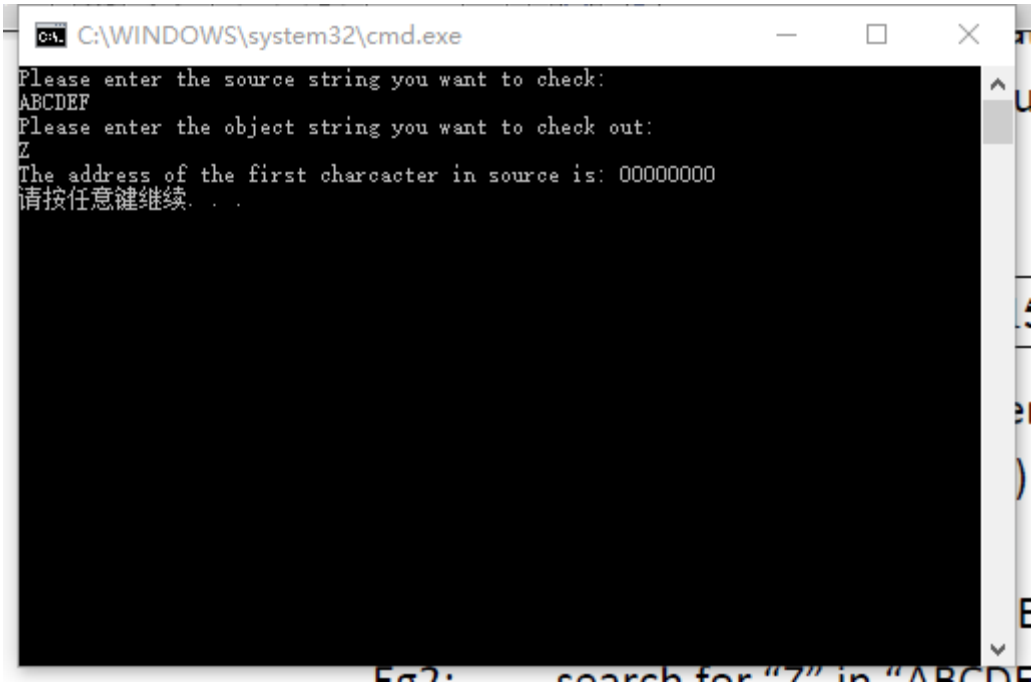


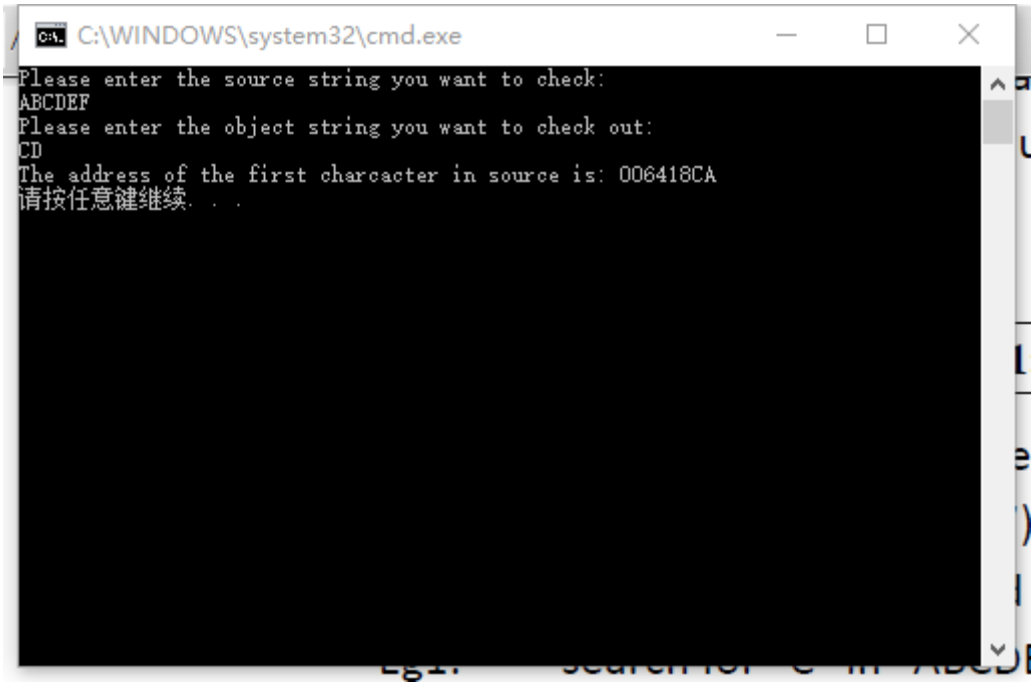
2. Self- test: to check if order or mutiplicities will affect the result.

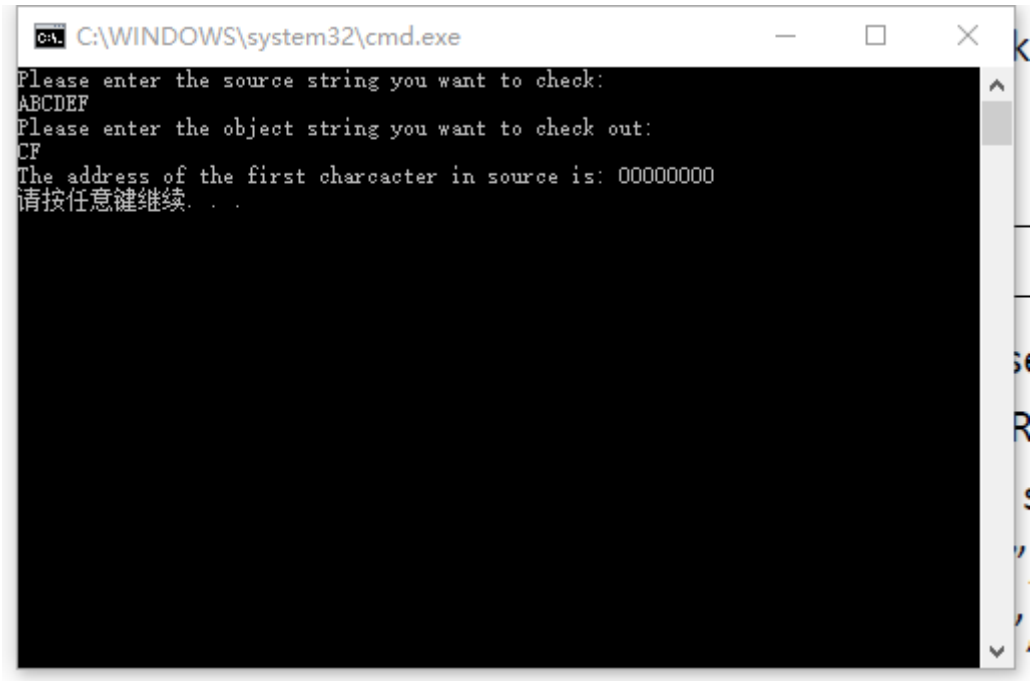
b) Exercise 2:

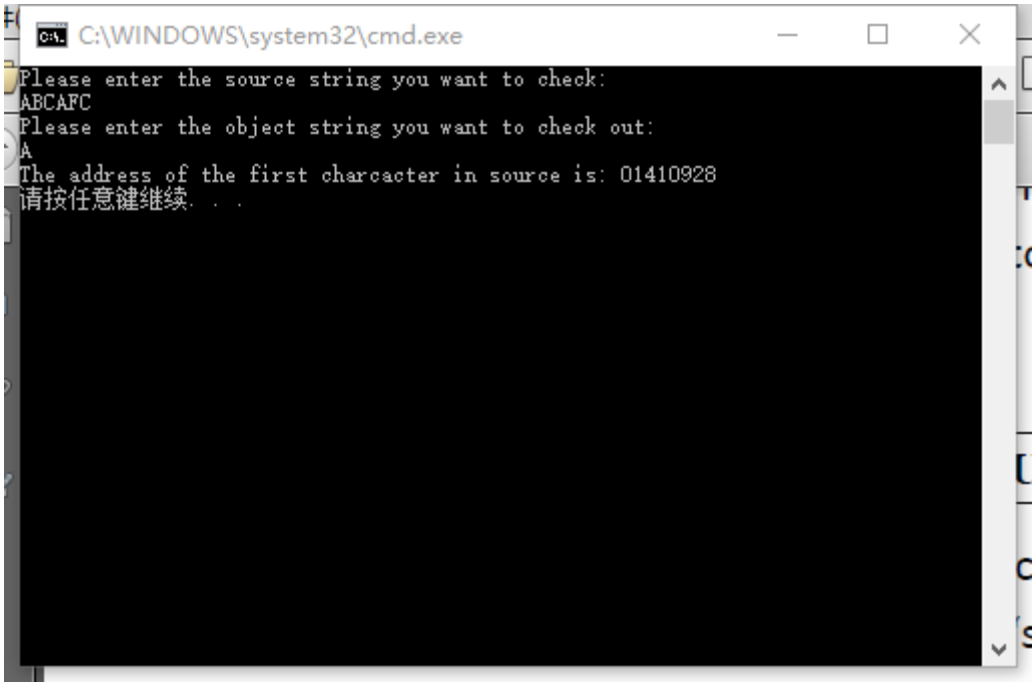
Commanded Test in Example:





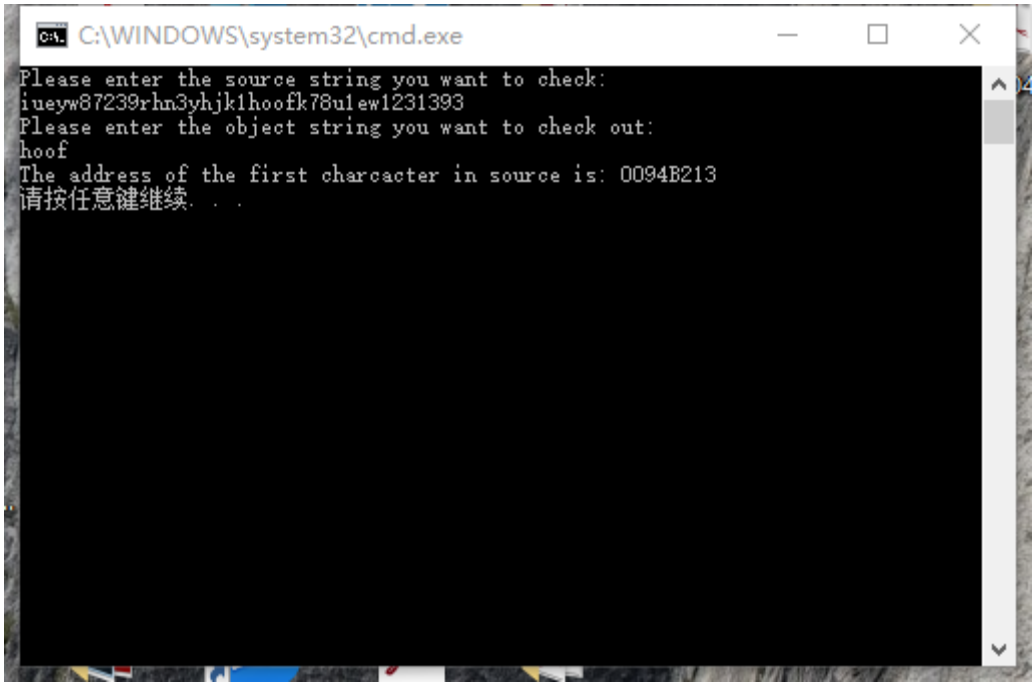

C:\WINDOWS\system32\cmd.exe
Please enter the source string you want to check:
ABCDEF
Please enter the object string you want to check out:
CD
The address of the first charcacter in source is: 006418CA
请按任意键继续. . .


C:\WINDOWS\system32\cmd.exe
Please enter the source string you want to check:
ABCDEF
Please enter the object string you want to check out:
CF
The address of the first charcacter in source is: 00000000
请按任意键继续. . .

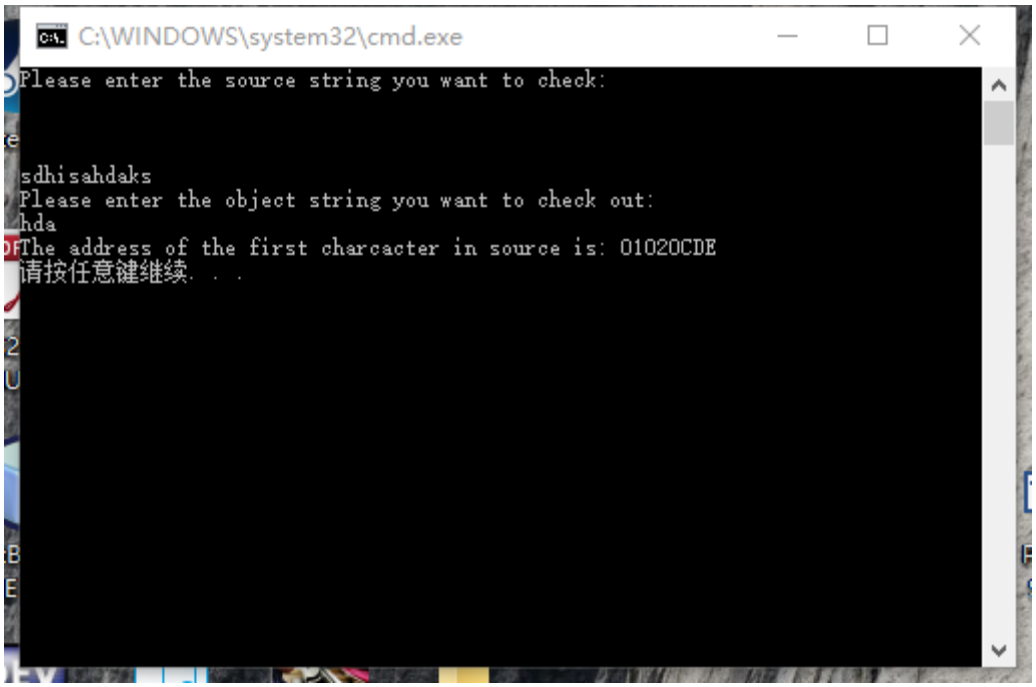


Self-test

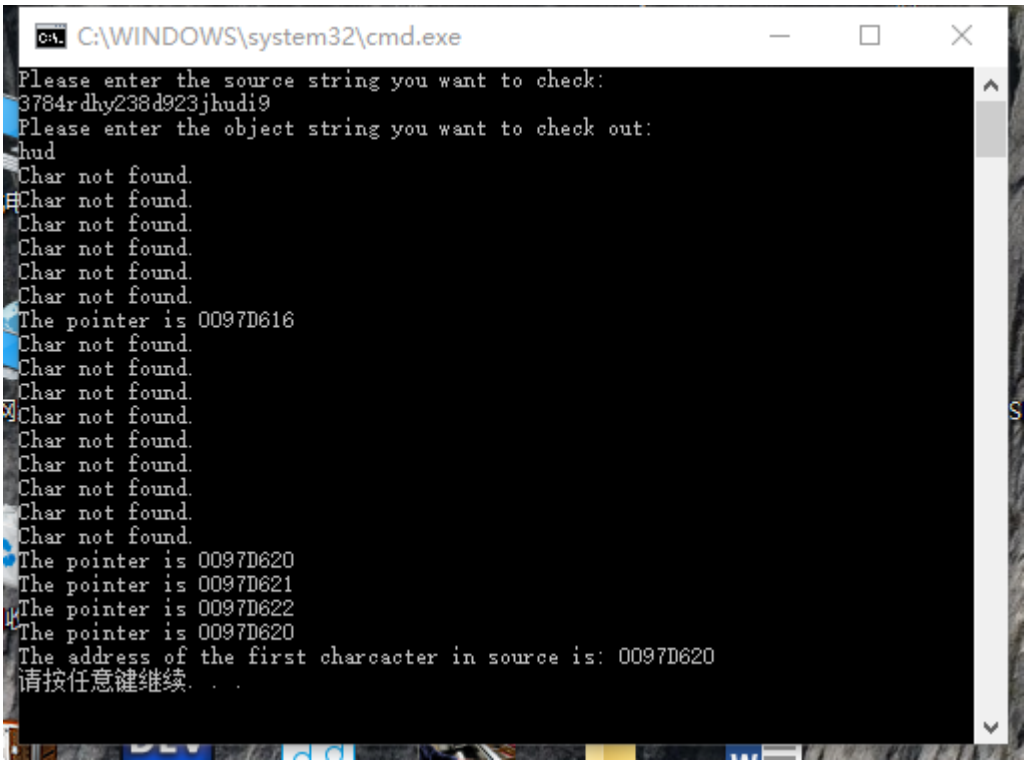
1. **To check if alpha-numeric combined input will affect result.**



2. **To check if erroneous newline input will affect result.**



3. **Further test for developers – achive by enable lines in source code**



```
int i, j , k;
//if the (remaining)source is not long enough to chceck out obj, terminate loop
if(strlen(source) >= strlen(obj)){
//IF ith element in source = jth element in obj, go on checking following elements
for (i = 0, j = 0; i < strlen(source) -strlen(obj)+1; i++) {
    if (source[i] == obj[j]) {
        k = i;  //copy i to k to protect the possile restart point
        //check elements follwing i and use count as a flag
        for (count = 0; source[k] == obj[j] && j < strlen(obj); k++, j++) {
            //cout << "The pointer is " << (int*)&source[k] << endl;
            count++;
            //If obj checked in source, return &source[i],
            //otherwise, set j = 0 prepare for restart check from source[i+1]
            if (count == strlen(obj)){
                ptr = (char*)&source[i];
                //cout << "The pointer is " << (int*)ptr << endl;
                i = strlen(source);
            }
        }
    }
    else{
        //cout << "Char not found.<<endl";
```