Implement string pattern matching using Rabin- Karp algorithm.

Given a text txt[0. . .n-1] and a pattern pat[0. . .m-1], I write a function pattern_search(char pat[], char txt[]) that prints all occurrences of pat[] in txt[]. I assume that n > m.

```
d = 256

def pattern_search(pat, txt, q):
    M = len(pat)
    N = len(txt)
    i = 0
    j = 0
    p = 0
    t = 0
    h = 1


    for i in range(M-1):
        h = (h*d) % q

    # Calculate the hash value of pattern and first window

    for i in range(M):
        p = (d*p + ord(pat[i])) % q
        t = (d*t + ord(txt[i])) % q

    # Slide the pattern over text one by one
    for i in range(N-M+1):
```

```
        # Check the hash values of current window of text and pattern if the hash values match then only check for characters one by one
        if p == t:
            # Check for characters one by one
            for j in range(M):
                if txt[i+j] != pat[j]:
                    break
                else:
                    j += 1



            if j == M:
                print("Pattern found at index " + str(i))
            else:
                print("Pattern not Found")



        if i < N-M:
            t = (d*(t-ord(txt[i])*h) + ord(txt[i+M])) % q



            if t < 0:
                t = t+q
while True:

    txt = input("Enter your String: ")
    print('The string you enter',txt)
    #txt = "Dibya is my Name"
    #pat = "a"
    pat= input("Enter what pattern do you want to find a character or a word: ")
```

```
print('You are searcing for ', pat)
# A prime number
q = 101

# Function Call
pattern_search(pat, txt, q)

# Code for While Loop
cont = input("Do you want to continue? yes/no > ")

while cont.lower() not in ("yes", "no"):
    cont = input("Do you want to continue? yes/no > ")
if cont == "no":
    print("Bye for now. See you soon.")
    break
```

In this case I put a drive code where system will ask to define a string from user then it will ask for pattern that user is haunting for.

It will also ask to continue the process and user can terminate the loop using no in prompt.