

KMP MultipleTimes #string

```
vector<int> constructTempArray(string pattern) {
    vector<int> lps(pattern.length());
    int index = 0;
    for (int i = 1; i < (int) pattern.length(); ) {
        if (pattern[i] == pattern[index]) lps[i] = index + 1, ++index, ++i;
        else {
            if (index != 0) index = lps[index - 1];
            else lps[i] = index, ++i;
        }
    }
    return lps;
}

void KMPMultipleTimes (string text, string pattern) {
    bool found = false;
    vector<int> lps = constructTempArray(pattern);
    int j = 0, i = 0;
    // i --> text, j --> pattern
    while (i < (int) text.length()) {
        if (text[i] == pattern[j]) ++i, ++j;
        else {
            if (j != 0) j = lps[j - 1];
            else ++i;
        }
        if (j == (int) pattern.length()) {
            cout << "found match at : " << (i - pattern.length()) << endl;
            j = lps[j-1];
            found = true;
        }
    }
    if (!found) cout << "not found" << endl;
}

int main() {
    string text, pattern;
    getline(cin, text);
    getline(cin, pattern);
    KMPMultipleTimes(text, pattern);

    return 0;
}
```