

4. Search and Replace

C++ Solution

```
#include <iostream.h>

int main () {
    char buffer[1025];          // To hold the text in, remember 1k is 1024 bytes,
                                // but you have to add the '\0' at the end.
    char letter;                // A temporary variable for character input.
    int i;                      // A counting variable for the for loop.

    // NB : No program introduction, if it is going to be used with // file
    // redirection then we need to avoid printing anything but // results to
    // cout.

    // Input the data.
    for (i = 0; i < 1025; i++) {
        cin.read(&letter,1);
        if (cin.eof()) { // If we've hit the end of the file.
            break;
        } else {
            buffer[i] = letter;
        }
    }
    buffer[i] = '\0';          // End the string.

    // Scan the data for 'chalk' and 'cheez'
    for (i = 0; i < 1025; i++) {
        if (buffer[i] == 'c') { // It might be one of the two.
            if (buffer[i + 1] == 'h' && buffer[i + 2] == 'a' && buffer[i + 3] == 'l' && buffer[i + 4] == 'k') { // If it's chalk
                buffer[i + 2] = 'e'; // Make it cheez.
                buffer[i + 3] = 'e';
                buffer[i + 4] = 'z';
            } else if (buffer[i + 1] == 'h' && buffer[i + 2] == 'e' && buffer[i + 3] == 'e' && buffer[i + 4] == 'z') { // If it's cheez
                buffer[i + 2] = 'a'; // Make it chalk.
                buffer[i + 3] = 'l';
                buffer[i + 4] = 'k';
            }
        }
    }

    // Output the data.
    i = 0;
    while (buffer[i] != '\0') {
        cout << buffer[i]; // Print the character out.
        i++; // Move to the next character.
    }
    return 0;
}
```

Marking Guidelines

Feature	Marks
Nothing extra outputted (program headers, etc.)	1
Allowance for less than 1024 bytes of data	2
Identifying "chalk" and "cheez" without using library functions	2
Replacing the words without using library functions	2
Intelligent use of loops	1