

```
1  /*@author Michał Pawłowski*/
2
3  #define _CRT_SECURE_NO_WARNINGS
4  #define _CRTDBG_MAP_ALLOC
5
6  #include <crtdbg.h>
7  #include <stdio.h>
8  #include <stdlib.h>
9  #include <time.h>
10 #include <stdbool.h>
11 #include <assert.h>
12 #include <string.h>
13
14
15 /** Funkcja kopiuje zawartość pliku do bufora
16 @date 2020-04-25
17 @param buffer wskaźnik na bufor
18 */
19 void openFile(char **buffer) {
20
21     FILE* pFile;
22     long lSize;
23     //char* buffer;
24     size_t result;
25
26     pFile = fopen("index.html", "rb");
27     if (pFile == NULL) { fputs("File error", stderr); exit(1); }
28
29     fseek(pFile, 0, SEEK_END);
30     lSize = ftell(pFile);
31     rewind(pFile);
32
33     *buffer = (char*)malloc(sizeof(char) * lSize);
34     if (*buffer == NULL) { fputs("Memory error", stderr); exit(2); }
35
36     result = fread(*buffer, 1, lSize, pFile);
37     if (result != lSize) { fputs("Reading error", stderr); exit(3); }
38
39     fclose(pFile);
40 }
41
42 /** Funkcja określa czy char należy do znaków zapisu rzymskiego
43 @date 2020-04-25
44 @param sign char do sprawdzenia
45 */
46 bool ifRome(char sign) {
47
48     switch (sign){
49     case 'I':
50     case 'V':
51     case 'X':
52     case 'L':
53     case 'C':
```

```
54     case 'D':
55     case 'M': return true;
56         break;
57     default: return false;
58         break;
59     }
60 }
61
62 /** Funkcja wypisuje wszystkie liczby rzymskie w formacie      ↗
    <z1><z2><z3>V<z4><z5><z6> z pliku
63 @date 2020-04-25
64 @param buffer wskaźnik na bufor
65 */
66 void printNum(char* buffer) {
67
68     char* v;
69     v = strchr(buffer, 'V');
70
71     if (v && ifRome(*(v - 3)) && ifRome(*(v - 2)) && ifRome(*(v - 1)) &&      ↗
        ifRome(*v) && ifRome(*(v + 1)) && ifRome(*(v + 2)) && ifRome(*(v + 3))) ↗
    {
72         printf("%c%c%c%c%c%c\n", *(v - 3), *(v - 2), *(v - 1), *v, *(v +      ↗
            1), *(v + 2), *(v + 3));
73         printNum(v + 4);
74     }
75 }
76
77 int main() {
78     {
79     char* buffer = NULL;
80     openFile(&buffer);
81     printf("Znaleziono liczby rzymskie:\n");
82     printNum(buffer);
83     free(buffer);
84     }
85     _CrtDumpMemoryLeaks();
86     return 0;
87 }
```