

```
1 // W7_1.cpp : Defines the entry point for the console application.
2 //   Działania z tablica tekstowa. Przekazania tablicy o dwóch indeksach
3 //   w argumentach funkcji na przykładzie wymiany dwóch wierszy
4
5 #include "stdafx.h"
6 #include <stdlib.h>
7 #include <memory.h>
8
9 #define MAX_LINES 10
10 #define MAX_LEN 5
11
12 #pragma warning (disable : 4996)
13
14 //prototypy funkcji globalnych
15 void GLOBAL_Crash(char *message_to_be_printed, void **tab);
16
17
18 //prototypy funkcji lokalnych
19 //tablica o dwóch indeksach str jest zadeklarowana w argumentach formalnych
20 //jako tablica wskaźników
21 static void print_str(char *str[], char *comment);
22
23 //tablica o dwóch indeksach str jest zadeklarowana w argumentach formalnych
24 //jako wskaźnik na wskaźnik
25 static void wymiana_wierszy(char **str, int first, int second);
26
27 int _tmain(int argc, _TCHAR* argv[])
28 {
29     char **str = NULL;
30     int i,j;
31     int symb = '0';
32     symb--;
33
34     //alokujemy pamięć dla tablicy wskaźników
35     str = (char **)calloc(MAX_LINES, sizeof(char *));
36     if(!str)
37         GLOBAL_Crash("błąd wydzielania pamięci", (void **)str);
38
39     for(i=0; i<MAX_LINES; i++)
40     {
41         //alokujemy pamięć dla każdego wiersza
42         str[i] = (char *)calloc(MAX_LEN, sizeof(char));
43         if(!str[i])
44         {
45             GLOBAL_Crash("błąd wydzielania pamięci", (void **)str);
46         }
47
48         //przypisujemy każdemu elementowi wiersza 0 - '0', wiersza 1 - '1',....
49         symb++;
50         if(symb > '9') //kod symbolu przekroczył kod '9' - zaczynamy ponownie od '0'
51             symb = '0';
52
53         for(j=0; j<MAX_LEN; j++)
54         {
```

```
55         str[i][j] = symb;
56     }
57 }
58
59 //----- ↗
60 //przekazywanie argumentow faktycznych dla tablicy o dwuch indeksach
61 //przy deklaracji w argumentach formalnych jako tablice wskaznikow jest ↗
62     takie same
63 //jak przy deklaracji jako wskaznik do wskaznika
64 //----- ↗
65 //drukujemy tablice
66 print_str(str, "tablca zrodlowa");
67 printf("podaj pierwszy i drugi indeksy dla wymiany\n");
68 scanf("%d%d", &i, &j);
69
70 //przestawienie wierszy o indeksach i, j
71 wymiana_wierszy(str, i, j);
72
73 //drukujemy tablice
74 print_str(str, "tablca po wymianie");
75
76 //zwalniamy pamiec
77 GLOBAL_Crash("OK", (void **)str);
78
79 return 0;
80 }
81
82 void wymiana_wierszy(char **str, int first, int second)
83 /*=====
84 Przestawia wierszy o indeksach first i second
85 Tablica o dwuch indeksach zadeklarowana jako wskaznik na wskaznik
86 =====*/
87 {
88     //sprawdzamy poprawnosc indeksow
89     if(!str || first < 0 || first >= MAX_LINES || second < 0 || second >= ↗
90         MAX_LINES)
91     {
92         //zwalniamy pamiec, alokowana dynamicznie
93         GLOBAL_Crash("blad danych", (void **)str);
94     }
95
96     char *str_row = str[first];
97     str[first] = str[second];
98     str[second] = str_row;
99 }
100 void print_str(char *str[], char *comment)
101 /*=====
102 Drukuje na monitorze zaglowek (comment) i tablice o dwuch indeksach
103 Tablica o dwuch indeksach zadeklarowana jako tablice wskaznikow
104 =====*/
105 {
106     int i,j;
```

```
107
108     printf("%s\n", comment);
109
110     for(i=0; i<MAX_LINES; i++)
111     {
112         for(j=0; j<MAX_LEN; j++)
113         {
114             printf("%c\t", str[i][j]);
115         }
116         printf("\n");
117     }
118 }
119
120
121 void GLOBAL_Crash(char *message_to_be_printed, void **tab)
122 {
123     printf("%s\n", message_to_be_printed);
124     printf("wprowadz Ctrl+Z dla zakonczenia i nacisni ENTER\n");
125     while(getchar() != EOF)
126         ;
127
128     //zwalniamy pamiec alkowana dynamicznie
129     if (tab)
130     {
131         //ile wierszy ma tablica?
132         size_t no_rows = _msize(tab) / sizeof(void *);
133         for (size_t i = 0; i < no_rows; ++i)
134         {
135             if (tab[i])
136             {
137                 free(tab[i]);
138                 tab[i] = NULL;
139             }
140         }
141
142         free(tab);
143     }
144
145     //przerywamy wykonanie zadania
146     exit(1);
147 }
148
149
150
```