

Programowanie strukturalne

- wykład 5

dr Piotr Jastrzębski

Tablice jednowymiarowe

Uzupełnienie

```
#include <stdio.h>
#include <stdlib.h>
#define ROZMIAR 2

int main()
{
    int tab[ROZMIAR] = {2,3,-2,0};
    for(int i=0;i<ROZMIAR;i++)
    {
        printf("%d\n",tab[i]);
    }
    printf("%d\n",tab[3]);
    return 0;
}
```

Wykonanie zależy od kompilatora i standardu języka C.

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int tabb5[(int)2.5]={3,3,3};
    return 0;
}
```

Przykład do przeanalizowania

```
#include <stdio.h>

int main()
{
    int tab[] = {2, -3, 5, 18, 5, 8, 12, 44};
    printf("%p %p %d\n", tab, &tab, *tab);
    int *wsk;
    wsk = tab + -3;
    printf("%d \n", *(wsk+=4));
    printf("%p \n", (wsk + 5));
    printf("%p \n", &(wsk--=-2)[-2]);
    printf("%d \n", (wsk+=-1)[2]);
}
```

Pamięć wirtualna

Pamięć wirtualna – mechanizm zarządzania pamięcią komputera zapewniający procesowi wrażenie pracy w jednym, dużym, ciągłym obszarze pamięci operacyjnej podczas, gdy fizycznie może być ona pofragmentowana, nieciągła i częściowo przechowywana na urządzeniach pamięci masowej. Systemy korzystające z tej techniki ułatwiają tworzenie rozbudowanych aplikacji oraz poprawiają wykorzystanie fizycznej pamięci RAM w systemach wielozadaniowych.

ASLR

ASLR (Address Space Layout Randomization) tłumaczony jest jako mechanizm losowego generowania lokalizacji alokacji pamięci wirtualnej.

Wykonywanie czynności przedstawionych na dalszych slajdach związanych z ASLR może oznaczać narażenie komputera na niebezpieczeństwo i podatność na ataki. Wykonanie tych działań nie jest zalecane i robione tylko na własną odpowiedzialność.

Exploit Protection

Funkcja Exploit Protection jest wbudowana w system Windows 10 w celu zabezpieczenia urządzenia przed atakami. Twoje urządzenie jest od razu skonfigurowane za pomocą ustawień ochrony, które są odpowiednie dla większości użytkowników.

[Ustawienia funkcji Exploit Protection](#)

[Dowiedz się więcej](#)

Wymuś losowe generowanie obrazów (obowiązkowa funkcja ASLR)

Wymuś relokację obrazów, które nie zostały skompilowane z użyciem przełącznika /DYNAMICBASE

Włączone domyślnie



Generuj losowo alokacje pamięci (funkcja ASLR „od dołu do góry”)

Generuj losowo lokalizacje alokacji pamięci wirtualnej.

Włączone domyślnie



Funkcja ASLR o wysokiej entropii

Zwiększ zmienność podczas używania ustawienia Generuj losowo alokacje pamięci (funkcja ASLR „od dołu do góry”).

Włączone domyślnie



Napisy (łańcuchy znaków)

Napisy

Napis - ciąg składający się z conajmniej jednego znaku. Znaki cudzysłowiu nie są częścią łańcucha.

Język C nie posiada typu `string`/łańcuchowego. Wszystkie napisy traktowane są jako tablice typu `char`. Ostatnim znakiem w tablicy jest znak `\0`.

Znak a napis

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    char a = 'q';
    char b[] = "q";
    return 0;
}
```

strlen a sizeof

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main()
{
    char nap1[] = "Hello World";
    char nap2[50] = "Hello World";
    printf("%d\n", sizeof nap1);
    printf("%d\n", strlen(nap1));
    printf("%d\n", sizeof nap2);
    printf("%d\n", strlen(nap2));
    return 0;
}
```

Tablica a wskaźnik

```
#include <stdio.h>
#define NAPIS "jakiś tekst"

int main()
{
    char tab[] = NAPIS;
    const char *wsk = NAPIS;
    printf("adres napisu %p\n", "jakiś tekst");
    printf(" adres tab: %p\n", tab);
    printf(" adres wsk: %p\n", wsk);
    printf(" adres NAPIS-u: %p\n", NAPIS);
    printf("adres napisu: %p\n", "jakiś tekst");
    return 0;
}
```

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    char nap1[] = "absddfvjskjf";
```

```
    char *nap2 = "oijefj";
```

```
    nap1[4] = 'M';
```

```
    *(nap1 + 7) = 'M';
```

```
    nap2[2]='3';
```

```
    return 0;
```

```
}
```

Kopiowanie napisu

```
#include <stdio.h>

int main()
{
    char * napis = "ab6sWR";
    char * kopia;
    kopia=napis;
    printf("%s\n",napis);
    printf("%p\n",napis);
    printf("%p\n",&napis);
    printf("%s\n",kopia);
    printf("%p\n",kopia);
    printf("%p\n",&kopia);
    return 0;
}
```


Wczytywanie napisów

- ▶ scanf <https://pl.wikibooks.org/wiki/C/scanf>
- ▶ gets <https://pl.wikibooks.org/wiki/C/gets>
- ▶ fgets <https://pl.wikibooks.org/wiki/C/fgets>

Wyświetlanie napisów

- ▶ `printf` <https://pl.wikibooks.org/wiki/C/printf>
- ▶ `puts` <https://pl.wikibooks.org/wiki/C/puts>
- ▶ `fputs` <https://pl.wikibooks.org/wiki/C/fputs>

Funckje łańuchowe

- ▶ `strlen` <https://pl.wikibooks.org/wiki/C/strlen>
- ▶ `strcat` <https://pl.wikibooks.org/wiki/C/strcat>
- ▶ `strncat` <https://pl.wikibooks.org/wiki/C/strncat>
- ▶ `strcmp` <https://pl.wikibooks.org/wiki/C/strcmp>
- ▶ `strcpy` <https://pl.wikibooks.org/wiki/C/strcpy>
- ▶ `strncpy` <https://pl.wikibooks.org/wiki/C/strncpy>
- ▶ `sprintf` <https://pl.wikibooks.org/wiki/C/sprintf>

Typ `wchar_t`

https://en.wikibooks.org/wiki/C_Programming/wchar.h

https://en.cppreference.com/w/c/language/string_literal

Bibliografia

- ▶ https://pl.wikipedia.org/wiki/Pami%C4%99%C4%87_wirtualna ,
dostęp online 30.03.2020.
- ▶ Stephen Prata, Język C. Szkoła programowania. Wydanie VI,
Wyd. Helion, 2016.