Tartalomjegyzék

1. args1.c

```
#include "prog1.h"
#include <stdio.h>

// irjuk ki argc erteket, majd
// irassuk ki a parancssori argumentumokat (mindet)

int main(int argc, string argv[])
{
    printf("argc: %d\n", argc);
    puts("");

    for (int i = 0; i < argc; ++i)
    {
        printf("%s\n", argv[i]);
    }

    return 0;
}
```

2. args2.c

```
#include "prog1.h"
#include <stdio.h>

// $ ./a.out
// hello world
// $ ./a.out Laci
// hello Laci

int main(int argc, string argv[])
{
    if (argc == 2)
    {
        printf("hello␣%s\n", argv[1]);
    }
    else
    {
        printf("hello␣world\n");
    }

    return 0;
}
```

3. exit_code1.c

```c
#include <stdio.h>

int main()
{
    return 3;
}
```

4. exit_code2.c

```
#include "prog1.h"
#include <stdio.h>

int main(int argc, string argv[])
{
    if (argc != 2)
    {
        printf("Hiba! Adj meg egy parametert!\n");
        return 1;
    }

    printf("hello %s\n", argv[1]);

    return 0;
}
```

5. exit_code3.c

```
#include "prog1.h"
#include <stdio.h>
#include <stdlib.h>

int main(int argc, string argv[])
{
    if (argc != 2)
    {
        printf("Hiba! Adj meg egy parametert!\n");
        exit(1);    // program befejeztetese az adott hibakoddal
    }

    printf("hello %s\n", argv[1]);

    return 0;
}
```

6. nevek1.c

```
#include "prog1.h"
#include <stdio.h>

int main()
{
    string nevek[4];
    nevek[0] = "Anna";
    nevek[1] = "Bea";
    nevek[2] = "Cecil";
    nevek[3] = "David";

    printf("%s\n", nevek[0][6]);     // Veszelyes muvelet! Csak pelda
        !

    return 0;
}
```

7. nevek.c

```c
#include "prog1.h"
#include <stdio.h>

int main()
{
    string nevek[4];
    nevek[0] = "Anna";
    nevek[1] = "Bea";
    nevek[2] = "Cecil";
    nevek[3] = "David";

    printf("%s\n", nevek[0]);
    printf("%c%c%c%c\n", nevek[0][0], nevek[0][1], nevek[0][2],
       nevek[0][3]);
    // printf("%c%c%c%c%d\n", nevek[0][0], nevek[0][1], nevek[0][2],
        nevek[0][3], nevek[0][4]);

    return 0;
}
```

8. string1.c

```
#include "prog1.h"
#include <stdio.h>

int main()
{
    string s = get_string("Input:␣");
    printf("Output:␣");

    for (int i = 0; s[i] != '\0'; ++i)
    {
        printf("%c␣", s[i]);
    }
    puts("");

    return 0;
}
```

9. string2.c

```c
#include "prog1.h"
#include <stdio.h>
#include <string.h>

int main()
{
    string s = get_string("Input:␣");
    printf("Output:␣");

    for (int i = 0; i < strlen(s); ++i)
    {
        printf("%c␣", s[i]);
    }
    puts("");

    return 0;
}
```

10. string3.c

```
#include "prog1.h"
#include <stdio.h>
#include <string.h>

int main()
{
    string s = get_string("Input:␣");
    printf("Output:␣");

    int hossz = strlen(s);
    for (int i = 0; i < hossz; ++i)
    {
        printf("%c␣", s[i]);
    }
    puts("");

    return 0;
}
```

11. string4.c

```c
#include "prog1.h"
#include <stdio.h>
#include <string.h>

int main()
{
    string s = get_string("Input:␣");
    printf("Output:␣");

    for (int i = 0, hossz = strlen(s); i < hossz; ++i)
    {
        printf("%c␣", s[i]);
    }
    puts("");

    return 0;
}
```

12. uppercase1.c

```c
#include "prog1.h"
#include <stdio.h>
#include <string.h>

// a kisbetuket alakitsuk at nagybetusse

int main()
{
    string s = get_string("Elotte:␣");
    printf("Utana:␣");

    for (int i = 0, hossz = strlen(s); i < hossz; ++i)
    {
        if (s[i] >= 'a' && s[i] <= 'z')
        {
            // nagybetusitsuk
            printf("%c", s[i] - 32);
        }
        else
        {
            printf("%c", s[i]);
        }
    }
    puts("");

    return 0;
}
```

13. uppercase2.c

```c
#include "prog1.h"
#include <stdio.h>
#include <string.h>
#include <ctype.h>

// a kisbetuket alakitsuk at nagybetusse

int main()
{
    string s = get_string("Elotte:␣");
    printf("Utana:␣");

    for (int i = 0, hossz = strlen(s); i < hossz; ++i)
    {
        // s[i]: aktualis karakter
        printf("%c", toupper(s[i]));
    }
    puts("");

    return 0;
}
```