

**ANSI C** NYELVEN**26-27. ÓRA****MyLIB függvények**

Az ANSI C függvénykönyvtárakhoz hasonlóan mi is készíthetünk saját függvényeket. Saját függvényeinket általában a főprogramtól elkülönült forrásfájlba helyezzük el, amely szintén .C típusú. A forrásfájlban megvalósított eljárások (void típusú függvények) és függvények prototípusait (használatukat leíró deklarációt) pedig egy ANSI C függvénykönyvtárhoz (például stdio.h) hasonlóan egy header fájlban helyezzük el. Függvényeink használata során a főprogram forrásfájljába ezt a .H fájlt hívjuk be egy #include hivatkozással.

**Egy lehetséges MyLIB megoldás**

A korábban elkészített *float veletlen(float ah, float fh)*; prototípussal megadott véletlen számokat generáló függvényt az alábbi módon helyezhetjük el egy saját készítésű függvénykönyvtárba:

1. MyLIB.c forrásfájl létrehozás
2. MyLIB.h header (definíciós, vagy prototípusokat tartalmazó) fájl létrehozása
3. Bemutató program forrásfájljának létrehozása (demo.c)
4. Főprogram és saját függvénykönyvtár fordítása/szerkesztése
5. Bemutató program futtatása

**MyLIB.c forrásfájl tartalma**

```
#include <stdlib.h>
#include <time.h>
#include "MyLIB.h"

float veletlen(float ah, float fh) {
    static int f = 0;
    float vel;

    if (f == 0) {
        srand((int) time(NULL));
        rand();
        f = 1;
    }

    vel = (float) rand() / (float) RAND_MAX;
    vel = ah + vel * (fh - ah);
    return (vel);
}
```

**MyLIB.h header/prototípus fájl tartalma**

```
float veletlen(float ah, float fh);
```

**Bemutató program (demo.c) forrásfájljának tartalma**

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

int main(void) {
    int i;

    printf("\n5 darab veletlen szam generalasa:");
    for (i=0; i<5; i++) {
        printf("\n%d. szam: %.3f", i+1, veletlen(10.0, 20.0));
    }
    return (0);
}
```

**Főprogram és saját függvénykönyvtár fordítása/szerkesztése**

```
>tcc\tcc demo.c MyLIB.c
```

## Bemutató program (demo.exe) futtatása

```
>demo
5 darab veletlen szam generalasa:
- 1. szam: 18.359
- 2. szam: 17.552
- 3. szam: 12.879
- 4. szam: 17.505
- 5. szam: 18.810
>
```

### Feladat

Készítsen programot (*veletlenszamok.c* néven), amely minimum 1, maximum 90 db valós számot generál. A valós számok a felhasználó által megadott tartományban kerüljenek létrehozásra, de a tartomány maximum a [10.0 - 50.0] zárt tartományban helyezkedjenek el. A program használja a korábban létrehozott ***rndfloat()*** és ***rndfloatarrunq()*** függvényeket. Az említett függvényeket helyezze el a MyLIB függvénygyűjteményben és ezt követően használja azokat.

### Megoldás

A MyLIB.c függvénykönyvtár forrásfájlját bővítenünk kell a megadott két függvénnyel:

```
#include <stdlib.h>
#include <time.h>
#include "MyLIB.h"

float rndfloat(float ah, float fh) {
    static int f = 0;
    float v;
    time_t t;

    if (f == 0) {
        t = time(&t);
        srand((int) t);
        rand();
        f = 1;
    }

    v = (float) rand() / (float) RAND_MAX;
    v = ah + v * (fh - ah);
    return v;
}

void rndfloatarrunq(float ah, float fh, float v[], int n) {
    int i, j, f;
    float vsz;

    i = 0;
    do {
        vsz = rndfloat(ah, fh);
        f = 0;
        for (j=0; j<i; j++) {
            if (v[j]==vsz) {
                f = 1;
            }
        }
        if (f==0) {
            v[i++] = vsz;
        }
    } while (i < n);
    return;
}
```

A MyLIB.h prototípus fájlt bővíteni kell a két függvény leírójával

```
float rndfloat(float ah, float fh);
void rndfloatarrunq(float ah, float fh, float v[], int n);
```

A felhasználótól történő adatbeolvasás miatt alkalmaznunk kell a **getint()** és a **getfloat()** függvényeket is, amelyekkel szintén bővítenünk kell a MyLIB.c és MyLIB.h fájlok tartalmát. Előzőbe a függvények definícióit, amíg az utóbbiba a függvények deklarációit (prototípus leírásait) kell elhelyeznünk.

#### A MyLIB.c függvénykönyvtár bővítése

```
void baddata(void) {
    printf("Hibas adat, adj meg ujra!");
    return;
}

float getfloat(char s[], float ah, float fh) {
    float value;

    do {
        printf("%s [%.2f-%.2f]: ", s, ah, fh);
        scanf("%f", &value);
        if (value<ah || fh<value) { baddata(); }
    } while (value<ah || fh<value);
    return (value);
}

int getint(char s[], int ah, int fh) {
    int value;

    do {
        printf("%s [%d-%d]: ", s, ah, fh);
        scanf("%d", &value);
        if (value<ah || fh<value) { baddata(); }
    } while (value<ah || fh<value);
    return (value);
}
```

#### A MyLIB.h prototípus fájlt bővítése

```
void baddata(void);
float getfloat(char s[], float ah, float fh);
int getint(char s[], int ah, int fh);
```

#### A *veletlenszamok.c* forrásprogram tartalma

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

int main(void) {
    int i;
    int n;
    float ah, fh, v[90];

    n = getint("Darabszam....", 1, 90);
    ah = getfloat("Also hatar....", 10.0, 50.0);
    fh = getfloat("Felso hatar...", ah, 50.0);

    rndfloatarrunq(ah, fh, v, n);
    printf("\nA szamok:\n");
    for (i=0; i<n; i++) {
        printf("\n%2d: %.4f", i+1, v[i]);
    }
    return (0);
}
```

#### A program futtatásának egy lehetséges eredménye

```
>veletlenszamok
Darabszam.... [1-90]: 5
Also hatar.... [10.00-50.00]: 11
Felso hatar... [11.00-50.00]: 23

A szamok:

1: 16.1985
2: 18.6028
3: 17.1057
4: 22.0215
5: 19.2836
```

## A MyLIB saját függvénykönyvtár teljes terjedelemben a 15-MyLIB.zip állományban található!

A MyLIB könyvtár függvényeinek prototípusai:

```
// Ellenorzott adatok bevitele

void baddata(void);
float getfloat(char s[], float ah, float fh);
int getint(char s[], int ah, int fh);

// Veletlenszamok generalasa

float rndfloat(float ah, float fh);
void rndfloatarr(float ah, float fh, float v[], int n);
void rndfloatarrunq(float ah, float fh, float v[], int n);

int rndint(int ah, int fh);
void rndintarr(int ah, int fh, int v[], int n);
void rndintarrunq(int ah, int fh, int v[], int n);
```