

Jazyk C – Náhodná čísla, barvy a pole

Matěj Hanke

25. září 2025

1 Náhodná čísla

Pro generování pseudonáhodných čísel využíváme:

- `srand()` - inicializace generátoru
- `rand()` - vrací číslo 0 .. `RAND_MAX`
- `rand() % (hornimez - dolnimez + 1) + dolnimez` - číslo v rozsahu

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

srand(time(NULL)); // inicializace generatoru

int x1 = rand() % 20;           // <0,19>
int x2 = rand() % 20 + 1;       // <1,20>
int x3 = rand() % 11 + 20;      // <20,30>
int x4 = rand() % 21 - 10;      // <-10,10>
int x5 = rand() % (30 - 10 + 1) + 10; // <10,30>
int x6 = rand() % (10 - (-5) + 1) - 5; // <-5,10>
```

2 Barvy v konzoli (Windows)

Barvu textu a pozadí lze nastavit příkazem `system("COLOR XY")`, kde:

- X = barva pozadí
 - Y = barva textu
- | | |
|---------------|----------------------|
| 0 = černá | 8 = šedá |
| 1 = modrá | 9 = světle modrá |
| 2 = zelená | A = světle zelená |
| 3 = tyrkysová | B = světle tyrkysová |
| 4 = červená | C = světle červená |
| 5 = purpurová | D = světle purpurová |
| 6 = žlutá | E = světle žlutá |
| 7 = bílá | F = jasně bílá |

Příklady

```
system("COLOR FC"); // bile pozadi, cervene pismo
system("COLOR 1E"); // modre pozadi, zlute pismo
system("COLOR 2F"); // zelene pozadi, bile pismo
```

3 Pole

Pole je datová struktura s prvky stejného typu. Indexování začíná od 0.

```
int absence[35]; // bez inicializace
int cisla[5] = {10,20,30,40,50}; // s inicializací
```

Práce s polem

```
for (int i=0; i<5; i++) {
    cisla[i] += 1; // zvýšení každého prvku
}
```

4 Funkce a práce s poli

Příklad funkce, která vygeneruje náhodné hodnoty do pole a vrátí průměr.

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

#define N 10

double naplnPole(int p[], int n) {
    int soucet = 0;
    for (int i=0; i<n; i++) {
        p[i] = rand() % 100;
        soucet += p[i];
    }
    return (double)soucet / n;
}

int main() {
    srand(time(NULL));
    int pole[N];
    double prumer = naplnPole(pole, N);

    printf("Prvky pole:\n");
    for (int i=0; i<N; i++) {
        printf("%d ", pole[i]);
    }
    printf("\nPrumer = %.2f\n", prumer);

    return 0;
}
```

5 Podmínky a barvy

Následující program vygeneruje čísla a podle velikosti změní barvu konzole.

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

int main() {
    srand(time(NULL));
    int x = rand() % 100;

    printf("Vygenerovane cislo: %d\n", x);

    if (x < 33) {
        system("COLOR 2F"); // zelene
        printf("Mala hodnota\n");
    } else if (x < 66) {
        system("COLOR 6F"); // zlute
        printf("Stredni hodnota\n");
    } else {
        system("COLOR 4F"); // cervene
        printf("Velka hodnota\n");
    }

    return 0;
}
```

6 Kombinace všeho

Program vytvoří pole náhodných čísel, spočítá maximum a minimum a podle výsledku změni barvu výstupu.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define N 15

int main() {
    srand(time(NULL));
    int pole[N], i, min, max;

    // naplneni pole
    for (i=0; i<N; i++) {
        pole[i] = rand() % 101; // <0,100>
    }

    // vypis pole
    printf("Pole: ");
    for (i=0; i<N; i++) {
        printf("%d ", pole[i]);
    }

    // hledani minima a maxima
    min = max = pole[0];
    for (i=1; i<N; i++) {
        if (pole[i] < min) min = pole[i];
        if (pole[i] > max) max = pole[i];
    }

    printf("\nMinimum = %d, Maximum = %d\n", min, max);

    // podminka na barvu
    if (max > 80) system("COLOR 4F"); // cervene
    else if (min < 20) system("COLOR 1F"); // modre
    else system("COLOR 2F"); // zelene

    return 0;
}
```