12. jednorozměrné pole (C)

* Homogenní strukturovaný datový typ
* Spojitá oblast operační paměti, první prvek je na nejnižší adrese a poslední na nejvyšší
* Jednotlivé prvky mají pevně danou velikost, která je u všech prvků pole stejná
* Hodnota pole je odkaz na místo v paměti, kde je pole uloženo
* Ke každé hodnotě v poli se dá přistupovat individuálně pomocí indexů (začínají od 0)
* Může být statické, nebo dynamické (nastavuje se pomocí funkce malloc)
* Patří zde i řetězce (pole znaků)
* Pole se do paměti ukládají dvěma způsoby:

1. Statické pole



Celé pole se ukládá v zásobníku

1. Dynamické pole

Pointer na pole je uložen v zásobníku, ale paměť je alokována v haldě

Způsoby deklarace:

* int pole[100] - vytvoří v paměti místo na pole o velikosti int \* 100
* int pole[5] = {1,2,3,4,5} - deklarace i inicializace zároveň
* int \*pole;
* Od C99 platí i int pole[n] // kde n je proměnná typu int
* Int pole [5] = {} - novinka v C23 - na každou pozici uloží 0 - **prázdná inicializace**

Používání:

* Nejčastěji se pole zapisuje i vypisuje v cyklu s řídící proměnou (for)

Int \*pole;

Int i;

srand(time(NULL));

pole = (int \*) malloc(sizeof(int) \* 5);

for( i=0 ; i<5 ; i++) {

\*(pole+i) = rand()%20+1; // načte do pole na index i náhodnou hodnotu z //intervalu <1;20>

//označený způsob zapisování se jmenuje pointerová aritmetika, jinak se dá použít //pole[i] = …

}

Pole struktur:

#include <stdio.h>

struct Employee {

char Name[20];

int employeeID;

int WeekAttendance[7];

};

int main() {

int i;

struct Employee emp[2]; // Deklarace pole struktur Employee o velikosti 2

for( i=0 ; i<2; i++) {

scanf("%s", emp[i].Name); // dá se napsat i takto \*(emp+i)->Name - (emp+i) je pointer tudíž se místo tečky musí napsat šipka

scanf("%d", &emp[i].employeeID);

for (int j = 0; j < 7; j++) {

scanf("%d", &emp[i].WeekAttendance[j]);

}

}

for (int i = 0; i < 2; ++i) {

printf("\nname: %s \nID: %d \nAttendance: ", emp[i].Name, emp[i].employeeID);

for (int j = 0; j < 7; ++j) {

printf("%d ", emp[i].WeekAttendance[j]);

}

}

return 0;

}

(deklarování proměnných uprostřed kódu je možné od standardu C99)

## Pole ve funkci

Statické pole

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Dynamické pole

Obsah obrázku text, snímek obrazovky, Písmo, software

Popis byl vytvořen automaticky

## Třízení

Bubble sort

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Quick sort

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky