

Programovanie

3. ročník – odborná prax

Ing. Daniela Kravcová

JEDNODUCHÉ A ZLOŽENÉ ÚDAJOVÉ TYPY

Údajové typy

Jednoduché údajové typy:

1. char,
2. int,
3. float,
4. double

Zložené údajové typy:

1. štruktúry,
2. pole, záznam, súbor,
3. uniony

Fyzické údajové štruktúry:

1. vektor,
2. zoznam

Abstraktné údajové štruktúry:

1. tabuľka
2. lineárny zoznam s koncovým prístupom
 - zásobník,
 - front,
3. lineárny zoznam s nekoncovým prístupom
 - Jednosmerný, obojsmerný, cyklický
4. strom

Jednoduché (neštruktúrované) údajové typy

1. **typ char** - slúži na uloženie jedného znaku - zaberie v pamäti 1 bajt čiže 8 bitov
2. **typ int** - slúži na uloženie celého čísla - zaberie v pamäti väčšinou 2 byty čiže 16 bitov - teda čísla od -32768 po 32767
3. **typ float** - slúži na uloženie reálneho čísla - pre uloženie sa ráta s veľkosťou 32 bitov, číslo je rozdelené na mantisu a exponent
4. **typ double** - slúži na uloženie reálneho čísla - podobne ako float, ale s dvojnásobnou veľkosťou.

Štruktúrované údajové typy

1. **pole** - vzniká zložením **prvkov rovnakého** neštruktúrovaného alebo štruktúrovaného typu nachádzajúcich sa v **operačnej pamäti**
2. **záznam** - vzniká zložením **prvkov rôzneho** neštruktúrovaného alebo štruktúrovaného typu nachádzajúcich sa v **operačnej pamäti**
3. **súbor** - ide o ľubovoľne dlhú, ale konečnú, postupnosť bajtov ukončenú bajtom CTRL-D a uloženú na **vonkajšom pamäťovom médiu**

Pole

Zápis	Popis deklarácie
<code>int cisla[100];</code>	pole 100 celých čísel, čiže v pamäti sa vyhradí 100 prvkov typu int uložených za sebou
<code>char retazec[30];</code>	pole 30 znakov, čiže v pamäti sa vyhradí 30 prvkov typu char uložených za sebou
<code>struct game g[5];</code>	pole 5 prvkov typu struct game uložených v pamäti za sebou

K jednotlivým prvkom poľa sa dostaneme cez tzv. **index poľa**.

Ide o zápis, ktorý sa uvádza za menom poľa v zložených zátvorkách. Index je prirodzené číslo a začína od 0.

Záznam

Záznamy delíme na:

- **enumeračné záznamy**
- **štruktúry**
- **uniony**

- Na vytvorenie enumeračného záznamu sa používa kľúčové slovo **enum**.
- Na vytvorenie štruktúry sa používa kľúčové slovo **struct**.
- Na vytvorenie unionu sa používa kľúčové slovo **union**.

Súbor

Aby sme mohli pracovať so súbormi, musíme k nim najprv vytvoriť prístup:

Pre tieto účely bol vytvorený typ **FILE**, ktorého definícia sa nachádza v hlavičkovom súbore **stdio.h**.

Funkcie pre prácu so súbormi:

- **FILE* fopen (char* meno, char* mode)**
- **int fread (char* buffer, int size, int count, FILE* fp)**
- **int fwrite (char* buffer, int size, int count, FILE* fp)**
- **fclose (FILE* fp)**
- **fprintf (FILE* fp, char* format,)**
- **fscanf (FILE* fp, char* format,)**

Uniony

- **union** – je v podstate obyčajná štruktúra, kde všetky jej položky zdieľajú jedno pamäťové miesto, teda sa prekrývajú (sú na rovnakej adrese).
- Zatiaľ čo u obyčajných štruktúr bolo pre každú položku alokované zvláštne miesto v pamäti, v prípade unionu je teda alokované miesto iba pre najväčší z nich.
- **definícia unionu je prakticky zhodná s definíciou štruktúr**

Fyzická údajová štruktúra - vektor:

- **umožňuje sekvenčnú implementáciu údajových objektov v operačnej pamäti.**

Vlastnosti fyzickej údajovej štruktúry VEKTOR:

- jednotlivé položky sú umiestnené v operačnej pamäti za sebou
- k jednotlivým prvkom sa uplatňuje priamy prístup
- priamy prístup - jednoznačný algoritmus zistenia miesta položky na základe indexu (adresy)

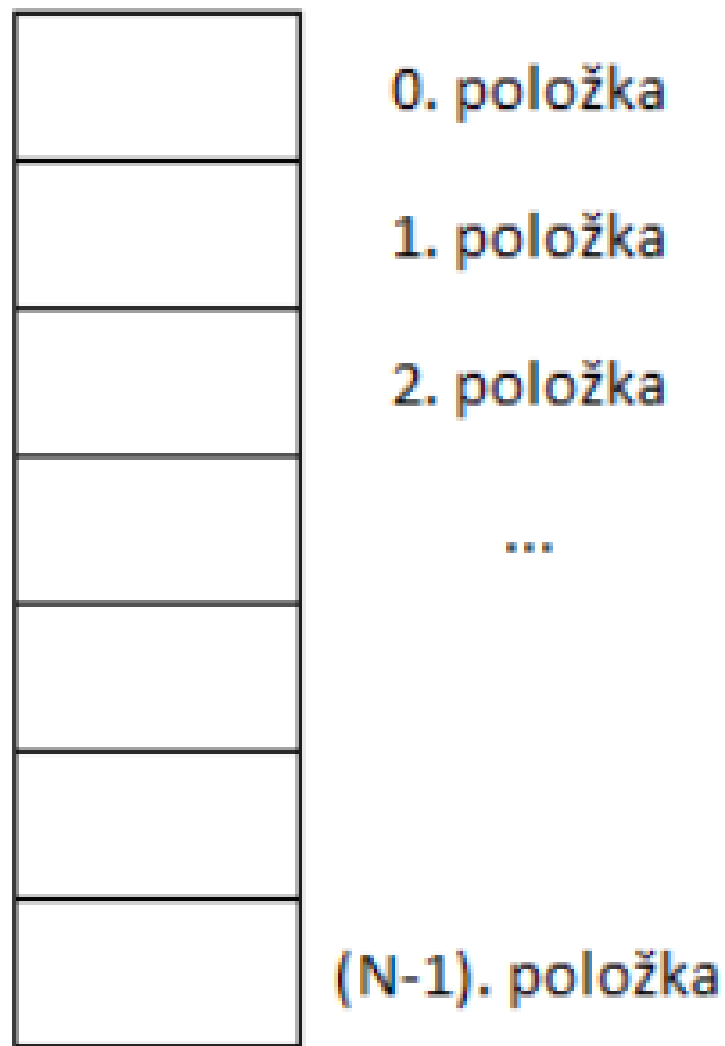
Statický vektor - nemení počet položiek v pamäti

- statický vektor má **VŽDY** obsadené všetky položky.

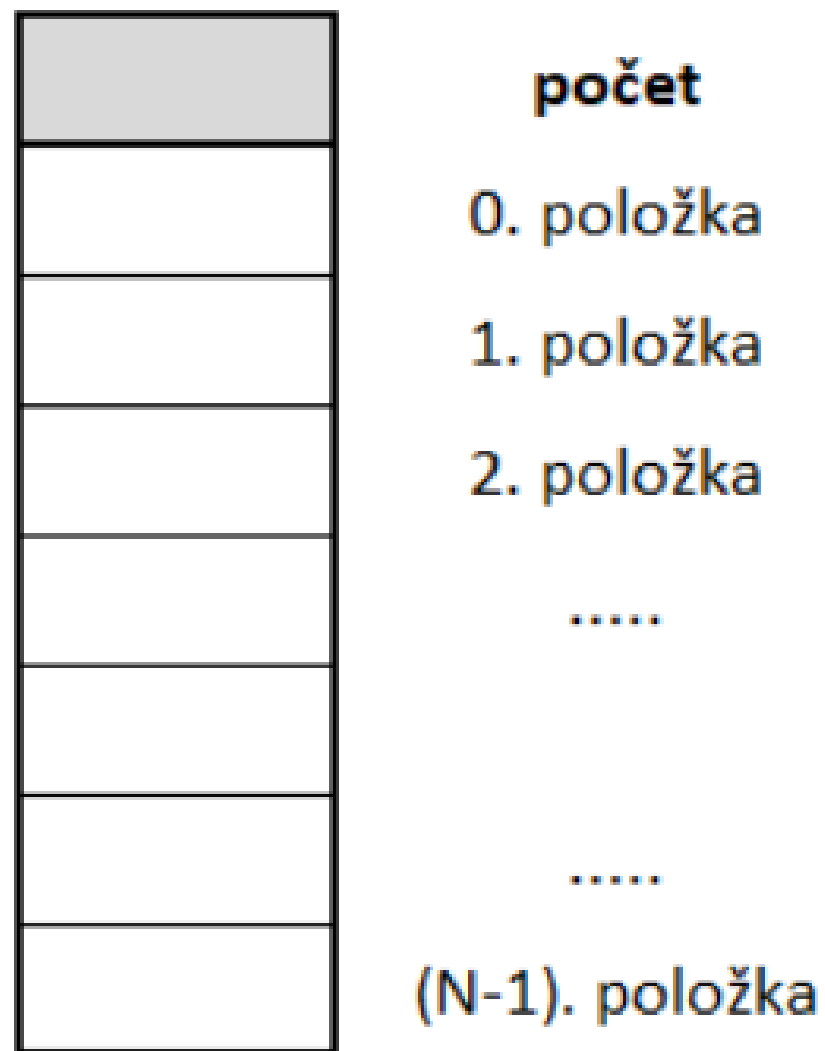
Dynamický vektor - mení počet položiek v pamäti

- nemusia byť obsadené všetky položky.
- o tom koľko položiek od začiatku dynamického vektora je obsadených hovorí údaj, ktorý je súčasťou štruktúry VEKTOR - údaj **počet**.

Grafické znázornenie



Statický vektor



Dynamický vektor

Fyzická údajová štruktúra - zoznam:

- **umožňuje spojkovú implementáciu údajových objektov v operačnej pamäti.**

Vlastnosti fyzickej údajovej štruktúry ZOZNAM:

- každá položka okrem informačnej časti nesie aj informácie o mieste výskytu iných (susedných) položiek

ZOZNAM-y poznáme:

- jednosmerný ZOZNAM
- obojsmerný ZOZNAM
- cyklický ZOZNAM
- viacsmerný ZOZNAM

Grafické znázornenie jednosmerného ZOZNAM-u

