

Pracovní list 2: Opakování algoritmizace II

Co už máme znát

- elementární algoritmy pro vyhledávání a řazení hodnot;
- způsoby zadání a načtení řady hodnot;
- pole;
- záznam (struktura);
- podprogram.

Kontrolní otázky

- 2.1 Jakými třemi způsoby lze zadat řadu hodnot na vstupu? Jaké algoritmy pro zpracování řady hodnot k jednotlivým způsobům patří?
- 2.2 V čem spočívá princip zpracování po sobě jdoucích dvojic hodnot vstupu?
- 2.3 K čemu slouží záznam a jak se definuje?
- 2.4 Jak se předávají parametry podprogramu? Co je reference (odkaz)?

Příprava na cvičení

Ve cvičení budeme potřebovat přístup k serveru akela, kde využijeme překladač jazyka C++, editor (joe, případně jiný) a schopnosti příkazového řádku.

Řešené příklady

Příklad 2.1 Napište podprogram, který přepočítá absolutní hodnoty vektoru desetinných čísel na hodnoty relativní vzhledem k maximu vyjádřené v procentech. Podprogram využijte pro vektor naplněný daty ze standardního vstupu.

Řešení: Podprogram pro zadanou úlohu bude procedura, jejímiž parametry budou naplněné pole desetinných čísel a počet naplněných složek pole. Vstupní pole bude přepočítáno a bude zároveň i výstupem procedury. Datový typ pole musí být definován ještě před definicí procedury, aby mohl být použit v její hlavičce.

```
28 | #include <iostream>
29 | using namespace std;
30 |
31 | const int MaxDelkaVektoru = 100;
32 | typedef double TypVektor[MaxDelkaVektoru];
```

```
33
34 void norma(TypVektor Vektor, int Obsazeno){
35     int KdeMax=0; //pozice maxima
36     for (int i=1; i<Obsazeno; i++) //nalezení pozice maxima
37         if (Vektor[i]>Vektor[KdeMax]) KdeMax=i;
38     for (int i=0; i<Obsazeno; i++) //přepočet
39         Vektor[i]=Vektor[i]/Vektor[KdeMax]*100;
40 }
41
42 int main(){
43     TypVektor V;
44     int Pocet=0;
45     while (cin>>V[Pocet] and Pocet<MaxDelkaVektoru)
46         Pocet++; //přečtení vstupu
47     norma(V, Pocet); //vyvolání procedury
48     for (int i=0; i<Pocet; i++)
49         cout << V[i] << endl; //výpis výsledku
50     return 0;
51 }
```

Příklad 2.2 Na vstupu se nachází na každém řádku dvojice údajů: identifikace pracovníka (trojznakový řetězec) a jeho plat. Vypište na výstup údaje o třech pracovnících s nejvyšším platem.

Řešení: Pro uložení údajů o pracovníkovi potřebujeme záznam se dvěma složkami. Protože potřebujeme uchovávat informace o třech pracovnících, bylo by vhodné k tomu vytvořit tříprvkové pole. Základní princip aplikovaného algoritmu spočívá ve zjednodušené podobě výběrového řadícího algoritmu v tomto trojprvkovém poli. Jednoduchou modifikací jediné konstanty tohoto programu můžeme získat obecně N pracovníků s nejvyšším platem, nebo dokonce seřadit celou skupinu podle platu sestupně.

```
52 #include <iostream>
53 using namespace std;
54
55 typedef struct {
56     char id[3];
57     int plat;
58 } TypPrac;
59
60 void zamena(TypPrac &A, TypPrac &B){
61     TypPrac Pom=A;
62     A=B;
63     B=Pom;
64 }
65
66 int main() {
```

```
67     const int PocNej = 3;
68     typedef TypPrac TypNejlepsi[PocNej+1];
69
70     TypNejlepsi Nejlepsi;
71     TypPrac Prac;
72     int PocPrac=0;
73
74     while (cin>>Prac.id) { //dokud není vše přečteno
75         cin>>Prac.plat; //přečtení údajů o pracovníkovi
76         if (PocPrac<=PocNej) { //ještě není přečten počet sledovaných
77             PocPrac++;
78             Nejlepsi[PocPrac-1]=Prac; //vložíme na konec
79         }
80         else Nejlepsi[3]=Prac; //jinak vkládáme do "rezervy"
81         for (int i=PocPrac-1; i>0; i--)
82             if (Nejlepsi[i].plat>Nejlepsi[i-1].plat)
83                 zamaena(Nejlepsi[i], Nejlepsi[i-1]); //umístění nového
84     }
85     for (int i=0; i<PocPrac-1; i++) //výpis výsledku
86         cout << Nejlepsi[i].id << " " << Nejlepsi[i].plat << endl;
87     return 0;
88 }
```

Příklady

Příklad 2.3 Napište podprogram, který vypočte součet hodnot vektoru desetinných čísel.

Příklad 2.4 Napište podprogram, jehož výsledkem je pole naplněné celočíselnými hodnotami ze standardního vstupu.

Příklad 2.5 Na vstupu je na každém řádku údaj o názvu zboží (řetězec o délce nejvýše 20 znaků), cena za kus a počet kusů na skladě. Vypište na výstup seznam zboží s nadprůměrným množstvím kusů na skladě. Dále vypište seznam všech názvů zboží, jehož množství na skladě je menší než 5 kusů.

Příklad 2.6 Standardní vstup obsahuje řadu desetinných čísel. Vypište tato čísla na výstup seřazená sestupně metodou přímého vkládání. Pro samotné seřazení vytvořte podprogram.

Příklad 2.7 Výměnou řadicího podprogramu z předchozí úlohy realizujte řazení přímým výběrem.

Co máme po cvičení umět

- Použití záznamu (**struct**).

- Definice podprogramu a volba způsobu předání parametrů.
- Využití záznamu, pole.
- Aplikovat jednoduché řadicí metody nebo jejich části (Bubble sort, přímý výběr, přímé vkládání).

Kontrolní otázky

- 2.5 Jak se liší zápis deklarace proměnné typu záznam a definice datového typu záznam?
- 2.6 Jak pracuje řadicí metoda Bubble sort?
- 2.7 Jaký je princip řadicí metody přímého výběru a přímého vkládání?
- 2.8 Kdy je výhodné vytvořit funkci a kdy naopak proceduru s jedním výstupním parametrem?
- 2.9 Jak se liší předání parametru některého jednoduchého typu a typu pole?
- 2.10 Je-li vstupem nebo výstupem podprogramu pole, jakou informaci je obvykle nutné ještě předávat?
- 2.11 Napište hlavičku procedury se zvoleným názvem, jejíž první celočíselný parametr se předává hodnotou a druhý celočíselný parametr odkazem.