



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií



Semestrální práce číslo 3

Autor práce: Andrii Diadius

Cíl práce:

Mým cílem bylo vytvořit kód,
který by načítal dvě posloupnosti
a rozhodl, zda první posloupnost
obsahuje druhou.

Komunikace s uživatelem
by měla vypadat tak:

```
Pokracovat ve zpracovani (a/n):  
a  
Zadej posloupnost:  
0 1 0 1 2 4 8 12 65 1 2 0 1 1 6 -1  
Hledana sekvence:  
0 1 2 -1  
Sekvence nalezena na pozici 3  
  
Pokracovat ve zpracovani (a/n):  
Y  
Pokracovat ve zpracovani (a/n):  
Y  
Pokracovat ve zpracovani (a/n):  
A  
Zadej posloupnost:  
7 0 7 5 25 21 1 2 2 1 11 12 6 8 5 1 2 3 1 11 -1  
Hledana sekvence:  
1 2 3 2 1 -1  
Sekvence nenalezena  
  
Pokracovat ve zpracovani (a/n):  
N
```

Nalezené řešení:

Pro načítání posloupnosti
jsem použil obecné pole
které rozšiřoval vlastní metodou
v případě uvedení uživatelem
více prvků.

Potom pomocí dvou „fórů“
porovnával prvky dvou
posloupnosti

```
for (int q = 0; q < mainAr.length - 1; q++) {  
    tempi = q;  
    for (j = 0; j < searchedAr.length - 1; j++) {
```

Po nalezení sekvencé
program vypisuje výsledek
a pozice, na které byla
nalezena sekvence

```
Zadej posloupnost:  
0 9 8 7 6 5  
-1  
Hledana sekvence:  
9 8 7 -1  
Sekvence nalezena na pozici 2
```

Postupné vyřešení úlohy počítačem vypadá tak:

- 1) Zjistit, zda uživatel chce spustit program
- 2) Načíst od uživatele první posloupnost
- 3) V případě více prvku rozšířit ji o jednu posice
- 4) Načíst od uživatele druhou posloupnost
- 5) Začít procházení od nultých prvku obou posloupností
 - 1) V případě shody dvou prvků, pokračovat porovnání dalších prvků
 - 2) Nebyla-li nalezena sekvence, vrátit se k prvku první matice kde sekvence se začala hledat
 - 3) Byla-li nalezena sekvence, ukončit algoritmus hledání, a vypsát výsledek
- 6) Opakovaně zjistit, zda uživatel chce spustit program ještě jednou

Testování:

Číslo testu	Typ testu, popis vstupů	Očekávaný výsledek	Skutečný výsledek	Prošel (ano/ne)
1	Obecný postup podle nápovědy programu Vstup: 0 1 0 1 2 4 8 12 65 1 2 0 1 1 6 -1 0 1 2 -1	“ Sekvence nalezena na pozici 3 ”	“ Sekvence nalezena na pozici 3 ”	ano
2	Obecný postup podle nápovědy programu Vstup: 7 0 7 5 25 21 1 2 2 1 11 12 6 8 5 1 2 3 1 11 -1 0 1 2 -1	“ Sekvence nenalezena ”	“ Sekvence nenalezena ”	ano
3	Při volbě *Pokračovat ve zpracování (a/n):*, zvolit *ano* velkým písmenem	*pokračování programu*	*pokračování programu*	ano
4	Opakovaná otázka *Pokračovat ve zpracování (a/n):* v případě zadání uživatelem špatné odpovědi	*Pokračovat ve zpracování (a/n):*	*Pokračovat ve zpracování (a/n):*	ano
5	Ukončení programu velkým písmenem N	*Ukončení*	*Ukončení*	ano

Výsledky testů:

Test 1:

```
Pokracovat ve zpracovani (a/n):  
a  
Zadej posloupnost:  
0 1 0 1 2 4 8 12 65 1 2 0 1 1 6 -1  
Hledana sekvence:  
0 1 2 -1  
Sekvence nalezena na pozici 3
```

Test 3:

```
Pokracovat ve zpracovani (a/n):  
A  
Zadej posloupnost:  
|
```

Test 2:

```
Pokracovat ve zpracovani (a/n):  
a  
Zadej posloupnost:  
7 0 7 5 25 21 1 2 2 1 11 12 6 8 5 1 2 3 1 11 -1  
Hledana sekvence:  
0 1 2 -1  
Sekvence nenalezena
```

Test 4:

```
Pokracovat ve zpracovani (a/n):  
g  
Pokracovat ve zpracovani (a/n):
```

Test 5:

```
Pokracovat ve zpracovani (a/n):  
N  
BUILD SUCCESSFUL (total time: 6 minutes 18 seconds)
```