

Semestrální úloha č. 1 z předmětu Algoritmizace II

Dekódování/kódování znaků Morseovy abecedy

Studenti:

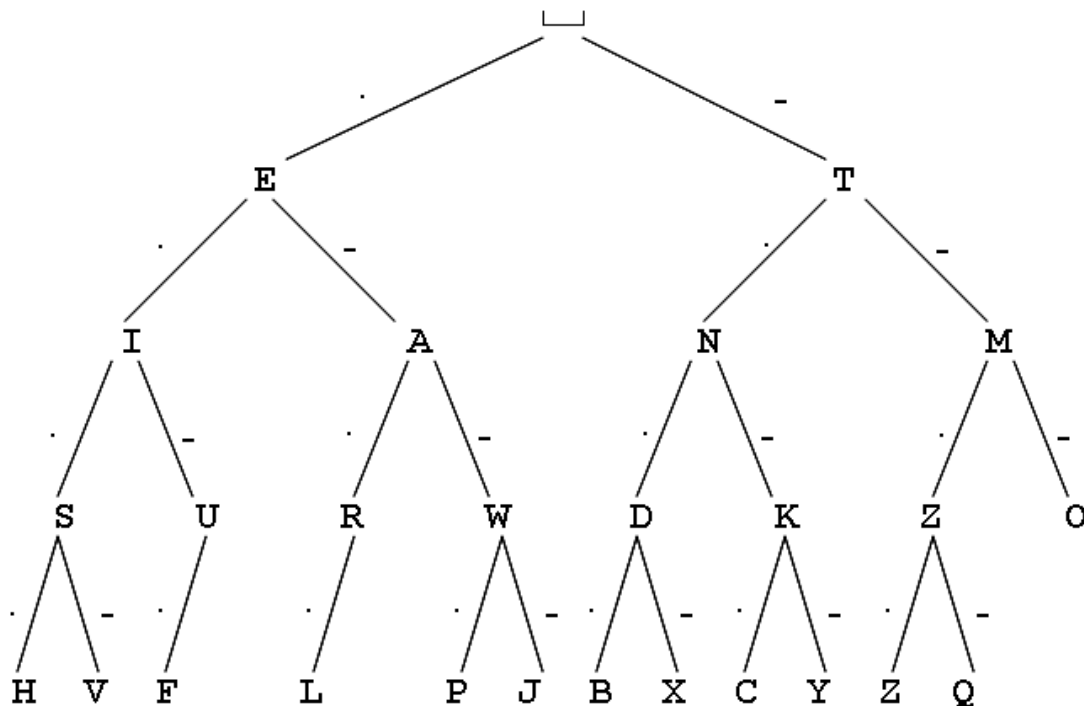
Naprogramujte překlad textových souborů z/do Morseovy abecedy. Překladovou databázi z Morseovy abecedy implementujte stromem; překlad do Morseovy abecedy polem řetězců. Při překladu do Morseovy abecedy je vstupem textový soubor s textem bez diakritiky, výstupem je textový soubor se znaky Morseovy abecedy oddělené / a naopak.

Příklad souboru s textem:

CAU

Příklad souboru s Morseovou abecedou:

-.-. / .- / ...-



Semestrální úloha 2 z předmětu Algoritmizace II

Výpočet nejkratší vzdálenosti v grafu Dijkstrovým algoritmem

Studenti:

Napište program, který vypočítá nejkratší vzdálenost mezi dvěma zadanými uzly v grafu Dijkstrovým algoritmem. Zvolte vhodnou reprezentaci grafu (např. maticí délek), kterou načtete z textového souboru, navrhněte reprezentaci situace, kdy mezi uzly není hrana. Využijte dynamickou alokaci paměti.

Příklad vstupního souboru (první řádek obsahuje počet uzlů grafu, tj. velikost matice, -1 znamená, že mezi uzly není hrana):

```
4
0 2 6 -1
2 0 3 2
6 3 0 5
-1 2 5 0
```

Semestrální úloha 3 z předmětu Algoritmizace II

Floydův-Warshallův algoritmus

Studenti:

Napište program, který implementuje Floydův-Warshallův algoritmus. Zvolte vhodnou reprezentaci grafu (např. maticí délek), kterou načtete z textového souboru, navrhnete reprezentaci situace, kdy mezi uzly není hrana. Využijte dynamickou alokaci paměti.

Příklad vstupního souboru (první řádek obsahuje počet uzlů grafu, tj. velikost matice, -1 znamená, že mezi uzly není hrana):

```
4
0 2 6 -1
2 0 3 2
6 3 0 5
-1 2 5 0
```

Semestrální úloha 4 z předmětu Algoritmizace II

Hledání minimální kostry algoritmem Jarníka

Studenti:

Napište program, který najde minimální kostru Jarníkovým algoritmem. Zvolte vhodnou reprezentaci grafu (maticí délek, spojovým seznamem uzlů), kterou načtete z textového souboru, navrhnete reprezentaci situace, kdy mezi uzly není hrana. Využijte dynamickou alokaci paměti.

Příklad vstupního souboru (první řádek obsahuje počet uzlů grafu, tj. velikost matice, -1 znamená, že mezi uzly není hrana):

```
4
0 2 6 -1
2 0 3 2
6 3 0 5
-1 2 5 0
```

Semestrální úloha 5 z předmětu Algoritmizace II

Hledání minimální kostry algoritmem Borůvky

Studenti:

Napište program, který najde minimální kostru Borůvkovým algoritmem. Zvolte vhodnou reprezentaci grafu (maticí délek, spojovým seznamem uzlů), kterou načtete z textového souboru, navrhnete reprezentaci situace, kdy mezi uzly není hrana. Využijte dynamickou alokaci paměti.

Příklad vstupního souboru (první řádek obsahuje počet uzlů grafu, tj. velikost matice, -1 znamená, že mezi uzly není hrana):

```
4
0 2 6 -1
2 0 3 2
6 3 0 5
-1 2 5 0
```

Semestrální úloha 6 z předmětu Algoritmizace II

Test acykličnosti a topologické uspořádání uzlů

Studenti:

Napište program, u zadaného orientovaného grafu zjistí, zda je acyklický a pokud ano, vypíše topologické uspořádání uzlů. Zvolte vhodnou reprezentaci grafu (maticí délek, spojovým seznamem uzlů), kterou načtete z textového souboru, navrhnete reprezentaci situace, kdy mezi uzly není hrana. Využijte dynamickou alokaci paměti.

Příklad vstupního souboru (první řádek obsahuje počet uzlů grafu, tj. velikost matice sousednosti):

```
4
0 1 0 1
1 0 1 1
1 0 0 0
0 0 1 0
```