

Referenčné riešenie - Tajemná zpráva

FIKS 2. kolo, 3. úloha, 9. ročník (2022/2023)

Úlohou bolo vytvoriť program vykonávajúci spätný beh programu v zadaní, ktorého funkciou bolo zakódovať správu (dlhšiu ako 2 znaky), ktorú obdrží na vstupe.

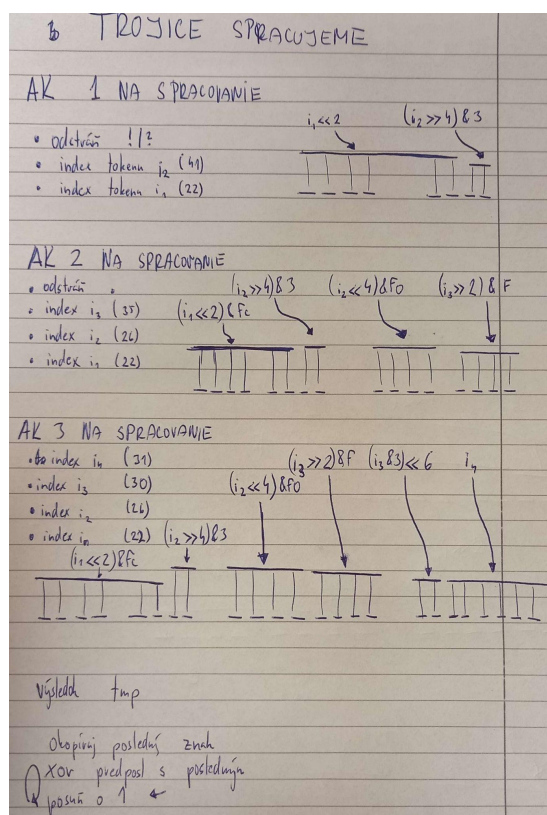
Pre vykonanie spätného behu bolo potrebné poriadne analyzovať kód vykonávajúci kódovanie.

Kódovanie sme rozdelili na 3 situácie:

- kódujeme 1 znak (vstup končí znakom !, alebo ?)
- kódujeme 2 znaky (vstup končí na znak .)
- kódujeme 3 znaky

Program kóduje text po trojiciach dokým je to možné a následne vyberie niektorú z prvých 2 možností pre dokončenie kódovania zvyšnej sekvencie. Kódovanie využíva operácie *bitový and* a *bitové posuny*, ktoré sú volené tak, že kódovanie je bezstrátové. (naviac kódovanie vykonáva pridávanie medzier, ktoré obsahujú iba redundantnú informáciu – je možné ju pre potreby dekódovania odstrániť)

Pre samotné dekódovanie správy je potrebné vyriešiť dekódovanie vyššie popísaných 3 situácií, ktoré sme na základe kódu popísali na papiery uvedenom nižšie:



Pred vykonaním samotného dekódovania bolo potrebné vykonať tokenizáciu kódovaného textu na základe kódovej abecedy, ktorú sme obdržali v rámci kódovacieho programu.

Tokenizáciu sme vykonali pomocou odoberania najdlhšej možnej slabiky obsiahnutej kódovej abecedy. Následovalo dekódovanie od posledných znakov kódovaného textu, to prebehlo na základe funkcie, ktorá vykonávala úpravy popísané na papieroch vyššie. Na záver bolo potreba vykonať opak počiatočnej úpravy kódovania, ktorá využíva XOR, ktorý je bezstrátový.

Náš program má časovú zložitosť $O(n)$.

Funkcia *tokenize* má časovú zložitosť $O(n)$ z reťazcu vždy zoberieme časť, ktorá predstavuje 1 slabiku kódovej abecedy a následne pôvodný reťazec prepíšeme skráteným reťazcom, čo sa vykoná za $O(1)$ prepisom dĺžky.

Funkcia *decode* pri každom priechode spracuje maximálne 3 slabiky pôvodnej abecedy a na nich vykonávame konštantne mnoho operácií. Následne spracovávame iba nespracované tokeny. Z toho vyplýva, že finálna časová zložitosť funkcie je $O(n)$.

Funkcie *xor* vykoná priechod reťazca a operáciu XOR. To je vykonané v čase $O(n)$ a nakoľko pri priechode cyklom ku každému prvku pristúpime 2-krát.

Všetky funkcie, ktoré sú súčasťou kódu fungujú v lineárnom čase, preto aj celý dekódér má časovú zložitosť $O(n)$.

Dešifrovaná správa je:

24.4.1719 Plavba zdala se byti nekonecnou. Nase obavy byly ale zazehnany kdyz jsme dnes spatrili zemi!

25.4.1719 Z cista jasna nas prekvapila boure a nahnala lod na skaliska u ostrova. Lod je poskozena a dalsi plavba vyzaduje opravy trupu. Nekolik dni zde budeme nuceni zustat.

27.4.1719 Ostrov je plny divych tvorů. Neboji se cloveka a utoci s ohromnou silou. Jen stezi jsem vyvazl z naseho setkani s nimi. Jeden z namorniku to stesti vsak nemel.

28.4.1719 Odhalil jsem duvod agresivity ostrovnich selem. Chrani sva cerstve narozena mladata!

30.4.1719 Po nekolika pokusech se nasi posadce podarilo spratelit se s touto divou zveri. Stacilo je nalakat na maso a bobule rostliny zvane fiksus.

1.5.1719 Opravy lodi jsou skoro hotovy. Jeste par dni a plavba bude moci pokračovat.

3.5.1719 V noci nas prekvapili nejaci lide. Netusime kdo to byl, nevideli jsme je. Zabili straze a vypalili tabor. My co jsme prezili se schovavame v jeskyni a doufame, ze nas nenajdou.