

Aritmetické kódování

Tomáš Maršálek

23. října 2011

1 Zadání

Naprogramujte v ANSI C přenositelnou konzolovou aplikaci, která jako vstup načte z parametru na příkazové řádce jméno souboru, který se má zkomprimovat/dekomprimovat, a jméno výstupního souboru, do kterého uloží zkomprimovaná/dekomprimovaná data. Program bude data komprimovat technikou aritmetické komprese (angl. Arithmetic Coding). Zda se bude provádět komprese nebo dekomprese se určí přepínačem na příkazové řádce.

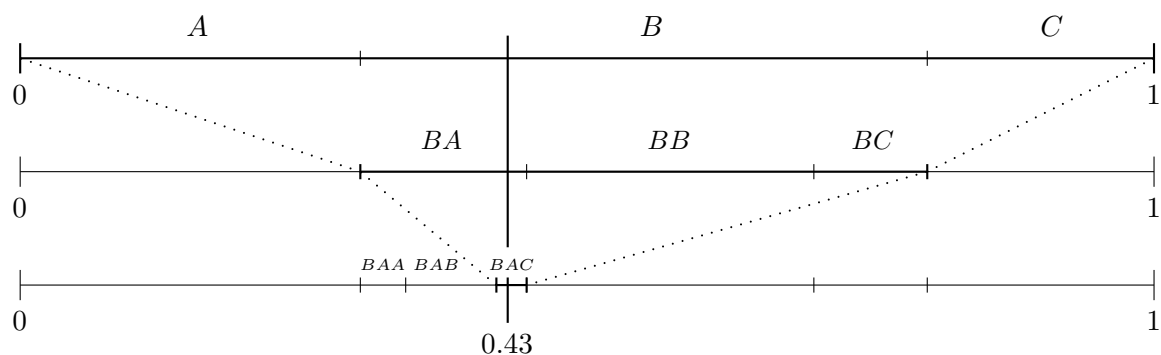
2 Analýza úlohy

Aritmetické kódování patří do kategorie takzvaných entropických kodérů. Ty ke kompresi využívají pravděpodobnostní rozdělení znaků řetězce. Příkladem je Morseova abeceda nebo Huffmanovo kódování. Obě tyto techniky přiřadí frekventovanějším znakům kratší kód a naopak delší méně frekventovaným. Tato kategorie kodérů je již vyřešený problém, protože Huffmanovo kódování přiřadí každému znaku optimální kód pro daný vstup. Teoretické maximum zakódování popisuje Shannonova věta o kódování bez šumu, podle ní je optimální délka $\log_2 \frac{1}{P}$ bitů na znak. P je pravděpodobnost výskytu znaku. Huffman, přestože určuje optimální kódy, teoretického minima dosáhne pouze pokud je pravděpodobnost znaku mocnina dvou, logaritmus pak vyjde celočíselně. Aritmetické kódování však funguje na jiném principu a díky tomu je možné znakům přiřadit i kódy neceločíselné délky. Zakóduje celou zprávu jako celek a tedy ve výsledku mohou mít jednotlivé znaky i zlomkové délky.

2.1 Princip

Vstup je mapován na číslo v intervalu $[0, 1)$. Interval je rozdělen na úseky odpovídající šířkou pravděpodobnostem znaků. Vstupní znak poté určí odpovídající subinterval, čímž o sobě zaznamená jednoznačnou informaci. Tento subinterval je rozdělen stejným způsobem podle pravděpodobností a další znak vybere svůj subinterval. Tímto způsobem se zakódují všechny znaky. Aritmetické kódování vlastně využívá nekonečné dělitelnosti množiny reálných čísel.

Příklad Pravděpodobnosti výskytu znaků jsou $A : 0.3$, $B : 0.5$, $C : 0.2$. Kódované slovo je BAC . V každém kroku je zvýrazněn právě vybraný interval a jeho rozdělení. Poté co určíme poslední subinterval, vybereme z něj libovolné číslo. To bude tvořit aritmetický kód zprávy.



Libovolné číslo z daného intervalu vybíráme s co nejkratší délkou, v tomto případě je kód zprávy 0.43.

Pro kratší kódy chceme, aby se vybíraly co nejdelší subinterval. Pak zůstane délka intervalu velká a nebudeme potřebovat tolik číslic k zapsání výsledného čísla. Proto je důležité mít správně zmapované pravděpodobnosti znaků, více frekventované znaky získají širší interval a tím pádem dochází ke kompresi. Kdyby měl program k dispozici informaci, že všechny znaky jsou stejně pravděpodobné, nedošlo by k žádné změně velikosti po zakódování (otestováno).

3 Implementace

4 Závěr