Imię i Nazwisko	Przedmiot	Data oddania	Ocena
Paulina Roszkowska Aleksandra Szum	Algorytmy i struktury danych	12 lutego 2020	
Sprawozdanie V			
Temat sprawozdania	Kodowanie Huffmanna		

1 Opis ćwiczenia

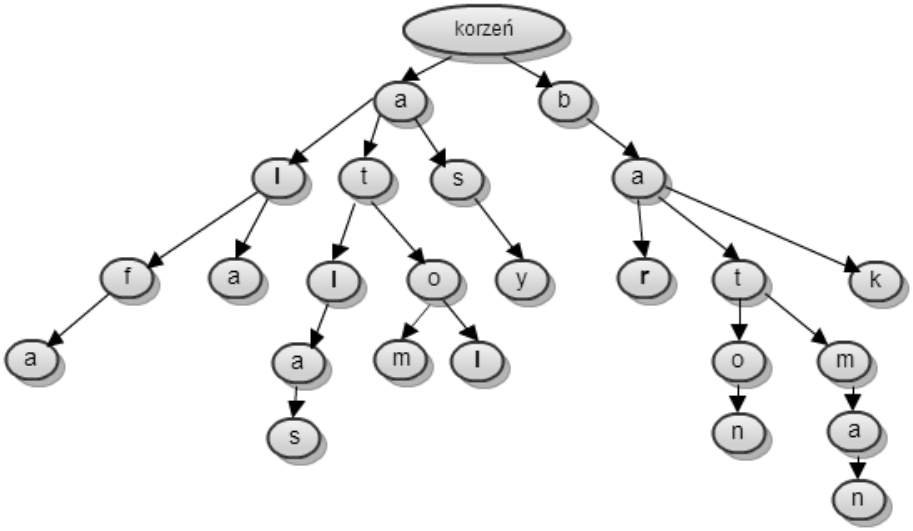
Ćwiczenie polegało na przeprowadzeniu kodowania Huffmana. Zatem wybrano dokument i dokonano kompresji tekstu metodą Huffmana. Opisano powstałe binarne drzewo prefiksowe, do każdej litery, spacji, nowej linii i znaków interpunkcyjnych). Zapisano otrzymane kody binarne. Dokonano kompresji 3 pierwszych zdań. Określono całkowity współczynnik kompresji na pełnym tekście.

2 Wstęp teoretyczny

Kodowanie Huffmana polega na przypisywaniu znakom występującym najczęściej w kodowanej wiadomości kodów krótkich. Znaki pojawiające się rzadziej otrzymują kody dłuższe. To zróżnicowanie długości pozwala otrzymać kod o mniejszej liczbie bitów, niż gdybyśmy stosowali kody o stałej długości.

Kod binarny reprezentuje tekst, instrukcje procesora komputera lub dowolne inne dane wykorzystujące system dwóch symboli. Używany system dwóch symboli to często „0” i „1” z systemu liczb binarnych. Kod binarny przypisuje wzór cyfr binarnych, znany również jako bity, do każdego znaku, instrukcji itp.

Współczynnik kompresji jest to procentowy udział zaoszczędzonego miejsca po spakowaniu pliku (zysk miejsca) w rozmiarze tego samego pliku przed jego kompresją. Zatem kompresja danych polega na zmianie sposobu zapisu informacji tak, aby zmniejszyć redundancję i tym samym objętość zbioru. Innymi słowy chodzi o wyrażenie tego samego zestawu informacji, lecz za pomocą mniejszej liczby bitów. Działaniem przeciwnym do kompresji jest dekompresja.



Rysunek 1: Przykładowe drzewo prefiksowe

Drzewo prefiksowe to w najprostszej postaci jest to drzewo, w którym każdy węzeł zawiera jeden znak (literę), oraz wskaźniki na swoje dzieci. Dzieci w każdym węźle może być tyle, ile liter w alfabecie.

Kolejne znaki położone na ścieżce od korzenia do liścia tworzą słowo.

Wyszukiwanie danego słowa w takim drzewie jest znacznie szybsze niż metoda porównywania szukanego słowa z całymi, pełnymi słowami umieszczonymi w tablicy.

Przykład drzewa prefiksowego dla listy słów został pokazany na rysunku 1: ala atom alfa baton atol batman bak asy atlas bar.

Uwagi dotyczące drzewa prefiksowego:

- mamy skończoną liczbę liter w alfabecie (np.: 26 – wszystkie małe litery alfabetu angielskiego)
- żadne słowo, które chcesz umieścić w drzewie, nie jest prefiksem innego słowa
- wyszukania słowa, które jest częścią jakiejś ścieżki, ale nie zaczyna się w korzeniu, będzie trudne, wymagające dodatkowych pomysłów

Możliwe problemy do rozwiązania z wykorzystaniem drzewa prefiksowego:

- Dodawanie słowa do drzewa
- Wyświetlenie listy znaków umieszczonych we wszystkich elementach drzewa
- Wyświetlenie listy wszystkich słów umieszczonych w drzewie
- Wyszukanie słowa w drzewie
- Autouzupełnianie słowa podawanego z konsoli po jednym znaku
- Usuwanie słowa z drzewa (trudne) – wymagałoby przechowywania dodatkowo w każdym węźle wskaźnika do jego ojca.

3 Obliczenia numeryczne

Kod znajduje się na stronie `https://github.com/aleksandraszum/AiSD_projekty/tree/master/report5`.

4 Analiza wyników

4.1 Kompresja 3 zdań

Pierwsze trzy zdania dokumentu brzmiały:

A Web Simulation of Medical Image Reconstruction and Processing as an Educational Tool

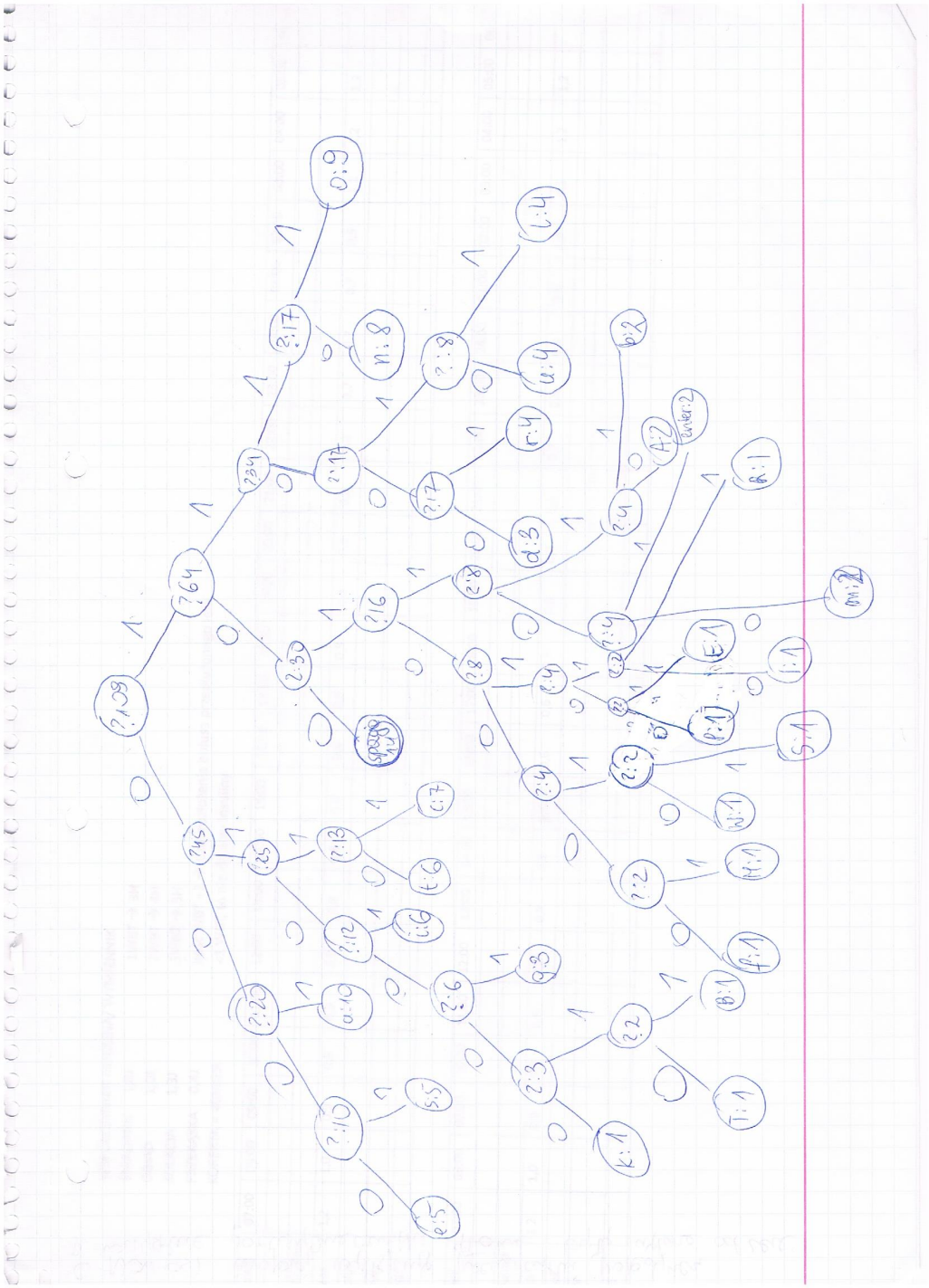
Abstract

Background

Dokonano na nich kodowania Hoffmana. Wyniki przedstawiono za pomocą drzewa prefiksowego oraz zbiorczej tabeli przedstawiającej kody binarne odpowiadające danemu znakowi.

4.1.1 Drzewo prefiksowe

Poniżej przedstawiono drzewo prefiksowe związane z kompresją pierwszych trzech zdań.



Rysunek 2: Utworzone drzewo prefiksowe związane z pierwszymi trzema zdaniami dokumentu

4.1.2 Kody binarne

W tabeli przedstawiono kody binarne znaków znajdujących się w pierwszych trzech zdaniach dokumentu.

Tabela 1: Znaki i odpowiadające im kody binarne.

Znak	Kod binarny	Znak	Kod binarny
e	0000	P	1010100
s	0001	E	1010101
a	001	I	1010110
k	010000	R	1010111
T	0100010	m	101100
B	0100011	enter	101101
g	01001	A	101110
i	0101	b	101111
t	0110	d	11000
c	0111	r	11001
spacja	100	u	11010
f	1010000	l	11011
M	1010001	n	1110
W	1010010	o	1111
S	1010011		

4.1.3 Zakodowane zdania

Poniżej przedstawiono zakodowane pierwsze trzy zdania tekstu:

101110 100 1010010 0000 101111 100 1010011 0101 101100 11010 11011 001 0110 0101 1111 1110 100 1111 1010000 100 1010001 0000 11000 0101 0111 001 11011 100 1010110 101100 001 01001 0000 100 1010111 0000 0111 1111 1110 0001 0110 11001 11010 0111 0110 0101 1111 1110 100 001 1110 11000 100 1010100 11001 1111 0111 0000 0001 0001 0101 1110 01001 100 001 0001 100 001 1110 100 1010101 11000 11010 0111 001 0110 0101 1111 1110 001 11011 100 0100010 1111 1111 11011 101101 101110 101111 0001 0110 11001 001 0111 0110 101101 0100011 001 0111 010000 01001 11001 1111 11010 1110

4.2 Całkowity współczynnik kompresji na pełnym tekście

Rozmiar pliku tekstowego: 7.26 KB
Rozmiar pliku po dokonaniu kompresji: 4.01 KB
Całkowity współczynnik kompresji:

$$W = \frac{4.01}{7.26} = 55.2\%.$$

5 Wnioski

Przeprowadzono kodowanie Huffmana na całym pliku tekstowym.

Trzy pierwsze zdania zostały przedstawione za pomocą drzewa prefiksowego i zbiorczej tabeli przedstawiającej znaki i odpowiadające im kody binarne.

Wyliczono całkowity współczynnik kompresji na pełnym tekście. Wynosi on 55.2%.

6 Bibliografia

1. Jerzy Wałaszek, *Kompresja Huffmana*, dostęp online: `https://eduinf.waw.pl/inf/alg/001_search/0121a.php`
2. Eduteka *Współczynnik kompresji* , dostęp online: `https://www.eduteka.pl/amp/doc/wspolczynnik-kompresji`
3. TopInfo, *Drzewo prefiksowe*, dostęp online: `https://sites.google.com/site/topinfo12/home/drzewo-prefiksowe`
4. Github, *Huffman coding*, dostęp online: `https://github.com/dileepnandanam/huffman-coding/blob/master/huff.py`
5. Github, *Huffman coding*, dostęp online: `https://github.com/bhrigu123/huffman-coding/blob/master/huffman.py`