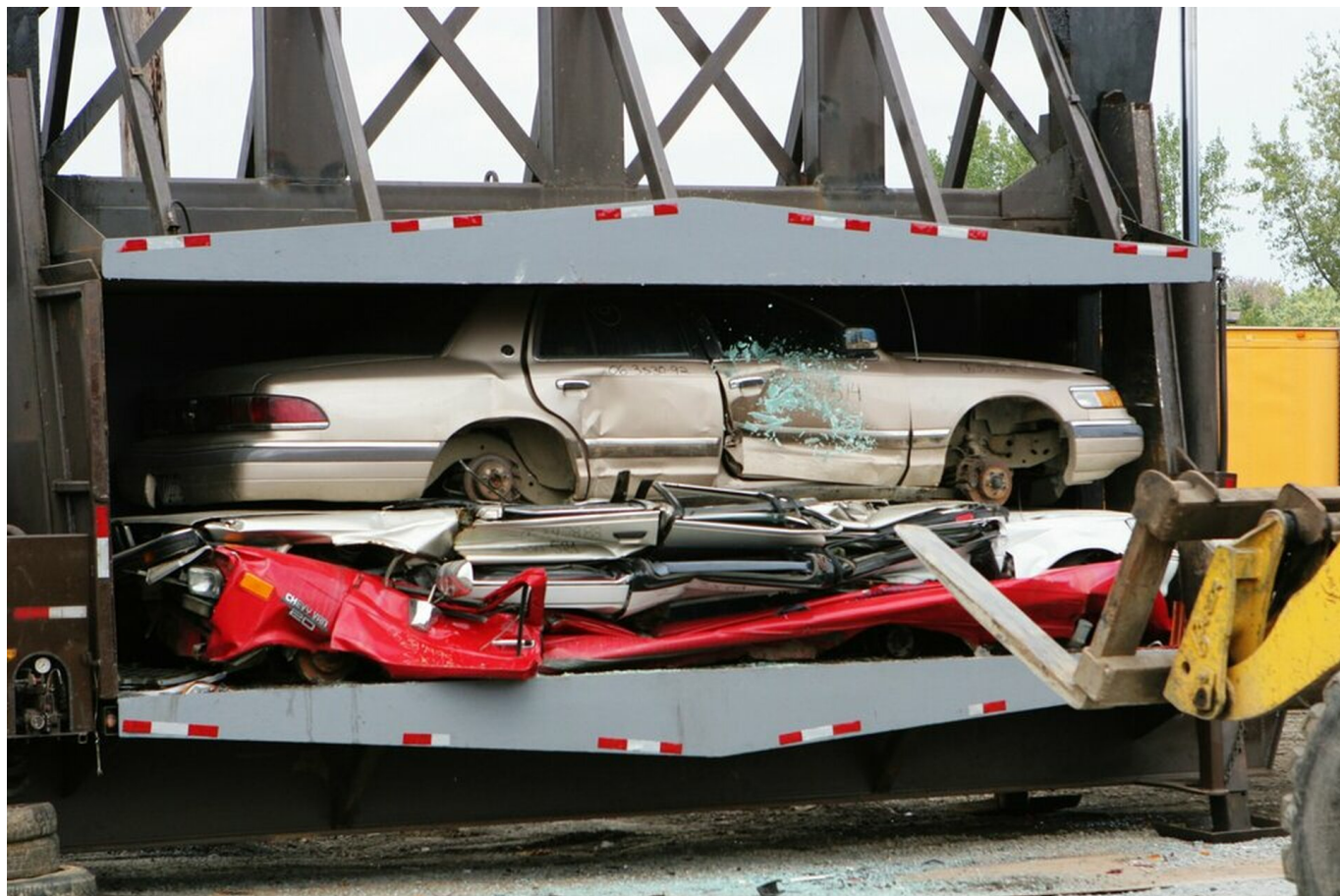




Kompresja danych

- [Wprowadzenie](#)
- [Przeczytaj](#)
- [Animacja](#)
- [Prezentacja multimedialna](#)
- [Dla nauczyciela](#)



Każda pobrana aplikacja na telefon, odtwarzany film w przeglądarce czy obrazek oglądany w internecie wymaga — w pierwszej kolejności — pobrania zasobu z internetu, a następnie otworzenia go. Bez kompresji do różnych formatów korzystanie z tych materiałów byłoby niemożliwe. Sprawdź, jakie są sposoby zapisywania informacji.

Więcej informacji dotyczących kompresji danych znajdziesz w pozostałych materiałach z serii:

- [Współczynnik kompresji danych](#),
- [Kompresja stratna i bezstratna](#),
- [Kompresja multimediów](#).

Więcej zadań? Zajrzyj do e-materiału: [Kompresja danych – zadania maturalne](#).

Twoje cele

- Wskażesz, jakie są sposoby zapisywania danych.
- Przeanalizujesz rodzaje kodowań używanych w informatyce oraz algorytmy kompresji.
- Wyjaśnisz, co to kod prefiksowy oraz czym różni się kompresja stratna od bezstratnej.

Przeczytaj

Sposoby zapisów informacji

Komputery korzystają z systemu binarnego, tzn. każda wartość, znak lub obrazek są złożone z ciągów liczb w [systemie dwójkowym](#).

O systemach liczbowych możesz przeczytać w e-materiale [Systemy liczbowe](#).

Kompresja danych

Kompresja jest to proces zmniejszania objętości, zagęszczenia lub wielkości czegoś. Można kompresować gazy, dane lub ciała stałe. W tym e-materiale interesuje nas kompresja danych w informatyce, tzn. zmiana sposobu zapisu informacji, tak aby zmniejszyć [redundancję](#) i tym samym objętość zapisanego zbioru.

Możemy zmniejszyć długość wiadomości na kartce przez zmianę sposobu kodowania.

Dany jest ciąg znaków, który nie zawiera cyfr.

Jeśli mamy ciąg znaków: „AAAAAAAAA”, możemy go zapisać w postaci: „9A” i dodać adnotację, że taki zapis oznacza dziewięć znaków „A”.

Definicja: Stopień kompresji

Miara względnego zmniejszenia rozmiaru reprezentacji danych generowanej przez algorytm kompresji danych (ang. *compression rate*, CR). Zwykle wyrażana jako wynik dzielenia rozmiaru nieskompresowanego przez rozmiar po skompresowaniu.

Dzięki temu udało nam się „skompresować” dziewięć znaków do dwóch, uzyskaliśmy stopień kompresji rzędu 4,5.

Stopień kompresji zapisuje się zgodnie z następującym wzorem:

$$\text{Stopień kompresji} = \frac{\text{Rozmiar nieskompresowanych danych}}{\text{Rozmiar danych po skompresowaniu}}$$

Definicja: Kompresja bezstratna

[Kompresja bezstratna](#) polega na zmianie kodowania wiadomości w taki sposób, że nie tracimy żadnych informacji. Operacja ta jest w pełni odwracalna, ponieważ polega jedynie na zmianie sposobu kodowania.

Kompresja bezstratna oznacza głównie zmianę sposobu kodowania wiadomości.

Przykład 1

Dana jest niezawierająca cyfr wiadomość: „AAAAAAAAAAAAAAAAAAAAAAAAAAAA”. Można skompresować ją poprzez zmianę kodowania do „24A” – co oznaczałoby, że została ona skrócona z 24 do 3 znaków.

Stopień kompresji wynosiłby więc: $24 : 3 = 8$.

Przykładem kodu, którym posługujemy się na co dzień podczas korzystania z komputerów, jest kod ASCII (skrót ang. American Standard Code for Information Interchange) oraz jego rozszerzenia. Kod ASCII koduje tylko na 7 bitach wszystkie litery alfabetu łacińskiego, języka angielskiego, znaki interpunkcyjne i znaki kontrolne.

Ciekawostka

ASCII jest kodem 7-bitowym, przy czym oryginalny standard nie definiuje roli ósmego bitu. Ósmy bit bywa wykorzystywany do kontroli parzystości lub do przechowywania dodatkowego atrybutu (np. podświetlenia). Najczęściej jednak ósmy bit służy rozszerzeniu podstawowego kodu ASCII o niezbędne znaki alfabetów narodowych, symbole matematyczne itp. Więcej na temat kodu ASCII znajdziesz w e-materiale [Reprezentacja tekstu w komputerze](#).

Ważne!

Co prawda zmiana sposobu kodowania jest niezbędna w kompresji, jednak sama nie powoduje oszczędności pamięci. Przykładowo przejście z kodu binarnego na kod Gray'a nie powoduje **żadnej** zmiany w ilości zajmowanej pamięci.

Kompresja przedstawiona w powyższym przykładzie, polega na zapisaniu informacji w inny sposób (w tym wypadku każdy zduplikowany znak występujący obok siebie możemy zastąpić liczbą, która oznacza, ile razy dany znak został powtórzony).

Definicja: Kompresja stratna

Polega na zmianie kodowania oraz na nieodwracalnym usunięciu części informacji. W kompresji stratnej niemożliwe jest odzyskanie informacji pierwotnej w postaci sprzed kompresji.

Algorytmy bezstratnej kompresji danych

Jest wiele algorytmów, które pozwalają bezstratnie kompresować dane. Przywołałyśmy jedynie kilka z nich.

Kodowanie Shannona

Kodowanie Shannona to metoda kompresji bezstratnej polegająca na zliczeniu częstości występowania danego ciągu liter lub poszczególnych znaków x_i w ciągu X , a następnie

posortowaniu ich w sposób nierosnący, zgodnie z prawdopodobieństwem P ich wystąpienia p_i .

Przykład 2

Dla dwudziestoelementowego ciągu „AAABBBBBBCCCCDDDDDDDD” tworzymy zbiór znaków: $X = \{A, B, C, D\}$. Następnie zliczamy wystąpienie każdego ze znaków:

$A : 3$

$B : 6$

$C : 4$

$D : 7$

Co pozwala nam policzyć prawdopodobieństwo p_i wystąpienia każdego ze znaków (zbiór składa się z 20 znaków):

$$p_A : \frac{3}{20} = 0,15$$

$$p_B : \frac{6}{20} = 0,3$$

$$p_C : \frac{4}{20} = 0,2$$

$$p_D : \frac{7}{20} = 0,35$$

Następnie sortujemy symbole wraz z częstościami nierosnąco:

$$X = \{D, B, C, A\}$$

$$P = \{0,35; 0,3; 0,2; 0,15\}$$

Tworzymy sumy prawdopodobieństw kolejnych znaków:

$$P_D = 0$$

$$P_B = p_1 = 0,35$$

$$P_C = p_1 + p_2 = 0,35 + 0,3 = 0,65$$

$$P_A = p_1 + p_2 + p_3 = 0,35 + 0,3 + 0,2 = 0,85$$

Obliczamy teraz długości Shannona, czyli długość kodów w bitach. Posłużymy się następującym wzorem:

$$l_i = \lceil -\log_2 p_i \rceil$$

Podstawiając kolejne prawdopodobieństwa, uzyskamy następujące wartości:

$$l_D = \lceil -\log_2 0,35 \rceil = \lceil 1,514573173 \rceil = 2$$

$$l_B = \lceil -\log_2 0,3 \rceil = \lceil 1,736965594 \rceil = 2$$

$$l_C = \lceil -\log_2 0,2 \rceil = \lceil 2,321928095 \rceil = 3$$

$$l_A = \lceil -\log_2 0,15 \rceil = \lceil 2,736965594 \rceil = 3$$

Zamieniamy prawdopodobieństwa z postaci dziesiętnej na binarną (prezentujemy pięć pierwszych bitów po przecinku):

$$P_D = 0,00_{10} = 0,00000_2$$

$$P_B = 0,35_{10} = 0,01011_2$$

$$P_C = 0,65_{10} = 0,10100_2$$

$$P_A = 0,85_{10} = 0,11011_2$$

Na koniec bierzemy z postaci binarnej pierwsze l_i bitów po przecinku – będzie to nasze słowo kodowe:

$$\text{kod}(A) = 0,11011 = 110$$

$$\text{kod}(B) = 0,01011 = 01$$

$$\text{kod}(C) = 0,10100 = 101$$

$$\text{kod}(D) = 0,00000 = 00$$

Wynikiem kodowania będzie następujący łańcuch znaków:

110 110 110 01 01 01 01 01 01 101 101 101 101 00 00 00 00 00 00 00₂

Kodowanie Huffmana

Kodowanie Huffmana to kolejna metoda bezstratnej kompresji danych. Jest to kodowanie prefiksowe polegające na zliczeniu częstości występowania danego ciągu liter lub znaków, które następnie będą kodowane zgodnie ze swoją częstością występowania. Więcej na temat tego kodowania możesz przeczytać w e-materiale [Kodowanie Huffmana](#).

Wiadomość „AAABBBBBBCCCCDDDDDDDD” po zastosowaniu kodowania Shannona przyjąłaby postać:

110 110 110 01 01 01 01 01 01 01 101 101 101 101 00 00 00 00 00 00 00₂

i miałyby długość 47 bitów. Ta sama wiadomość zgodnie z kodowaniem Huffmana wyglądałaby następująco:

110 110 110 10 10 10 10 10 10 111 111 111 0 0 0 0 0 0 0 0₂ i miałyby długość 37 bitów.

Aby zdekompresować wiadomość zapisaną zgodnie z kodowaniem Shannona, należy dopasować metodą prób i błędów słowa kodowe do odpowiednich fragmentów wiadomości.

Aby zdekodować wiadomość zakodowaną zgodnie z kodowaniem Huffmana, możemy posłużyć się zbudowanym **drzewem binarnym** i przechodzić po jego wierzchołkach zgodnie z kolejnymi bitami lub również próbować dopasowywać odpowiednie bity do słów kodowych.

Oczywiście możemy skonstruować drzewo również w kodowaniu Shannona, jednak posłużenie się nim byłoby działaniem dodatkowym, wymagającym kolejnych obliczeń.

Słownik

drzewo binarne

podtyp drzewa, w którym każdy wierzchołek ma nie więcej niż dwóch synów

kompresja bezstratna

kompresja, która pozwala po skompresowaniu odtworzyć dane wejściowe ze skompresowanej wiadomości

redundancja danych

nadmiarowość danych w bazie danych, które nie są konieczne do prawidłowego jej działania

system dwójkowy

(inaczej: binarny) system liczbowy, którego podstawą jest liczba 2, a do zapisu potrzebne są w nim tylko dwie cyfry: 0 i 1

Animacja

Ćwiczenie 1

Zapoznaj się z animacją. Spróbuj przeprowadzić eksperyment wraz z koleżanką lub kolegą.



Film dostępny pod adresem </preview/resource/Rq3nU6NmIpKpB>

Źródło: Contentplus.pl sp. z o.o., licencja: CC BY-SA 3.0.

Film nawiązujący do kompresji danych.

Polecenie 1

Opracuj notatkę podsumowującą najważniejsze informacje przedstawione w animacji.

Prezentacja multimedialna

Kodowanie arytmetyczne

Polecenie 1

Zapoznaj się z prezentacją przedstawiającą proces kodowania arytmetycznego.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

1

Wyobraźmy sobie, że całą wiedzę ludzkości chcemy zakodować na patyku, poprzez robienie na nim nacięć. Patek ten może mieć nawet jedynie metr długości. Teoretycznie pozwala na to kodowanie arytmetyczne, jednak prawa fizyki i dokładność odczytu oczywiście to uniemożliwiają.

Koncepcja algorytmu polega na dzieleniu pewnego odcinka na przedziały odpowiadające danej informacji. Każdy z przedziałów dzielimy następnie na kolejne przedziały - aż do zakodowania całej informacji. Jeżeli odpowiednio podzielimy patyk, to za pomocą pojedynczego nacięcia w odpowiednim z przedziałów jesteśmy w stanie (czysto teoretycznie) zapisać na nim dowolny zbiór informacji.

2

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Kodowanie arytmetyczne, podobnie jak poprzednie omawiane kodowania, wymaga

w pierwszej kolejności zliczenia częstości występowania danych (symboli/słów/znaków).

Następnie obliczamy prawdopodobieństwo wystąpienia każdej z tych danych i grupujemy je w kolejności od największego do najmniejszego prawdopodobieństwa.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

3

Dalej dzielimy przedział zgodnie z udziałem prawdopodobieństwa, od największego do najmniejszego. Jeżeli n to liczba przedziałów, a p_i , to prawdopodobieństwo wystąpienia danej informacji, otrzymamy następującą listę przedziałów:

$[\min, p_1), [p_1, p_1 + p_2), [p_1 + p_2, p_1 + p_2 + p_3), \dots, [p_1 + p_2 + p_3 + \dots$

Gdzie na początku $\min = 0$, a $\max = 1$.

4

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

W kolejnym kroku przechodzimy do kodowania wiadomości, wchodząc w coraz mniejsze przedziały wartości. Zagęszczanie przedziałów odbywa się poprzez ustalenie dolnej (min) oraz górnej (max) granicy nowego przedziału. Następnie ponownie dzielimy przedział jak w poprzednim kroku, w ten sposób, żeby każdy z przedziałów miał proporcjonalnie taką samą długość jak w podziale początkowym.



Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

5

Zagęszczanie i wchodzenie w kolejne przedziały kontynuujemy do momentu, aż zakończymy kodowanie całej wiadomości.

Przeanalizujmy omówiony proces na przykładzie.

6

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Przykład

Dla wiadomości

„AAAAAABBBBBBCCCCDDDD#” mamy następujące częstości:

- $A : 6$
- $B : 6$
- $C : 4$
- $D : 4$

W przypadku tego kodowania dla uproszczenia dodamy jeszcze znak #, który oznacza koniec kodowanej wiadomości. Jest on potrzebny, żeby w trakcie dekodowania uniknąć niejednoznaczności.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

7

Prawdopodobieństwo wystąpienia każdego ze znaków jest następujące:

- $A : \sim 0,286$
- $B : \sim 0,286$
- $C : \sim 0,190$
- $D : \sim 0,190$

„Pozostała część” wielkości $\# : 0,048$ oznacza koniec wiadomości.

Prawdopodobieństwa posłużą nam później do wyznaczenia końców przedziałów.

8

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

W praktyce najlepiej korzystać z jak najdokładniejszych przybliżeń, żeby uzyskać jak najlepsze wyniki. Jednak dla uproszczenia prawdopodobieństwo dla $\#$ zaokrągliliśmy do 0,048, aby prawdopodobieństwa poszczególnych symboli sumowały się do 1.

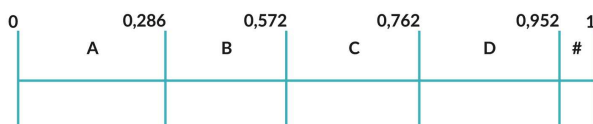
9

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Następnie na naszym „wirtualnym patyku” zaznaczamy prawdopodobieństwa wystąpienia kolejnych znaków w następujący sposób:

10



Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Każda litera ma swój własny „przydział” miejsca. Przedział dla litery A ma granicę od 0 do końca swojego prawdopodobieństwa, czyli 0,286. Przedział od B ma granicę wyznaczoną zgodnie z następującą zasadą: „Weź koniec poprzedniego przedziału (0,286) – to początek – i dodaj swoje prawdopodobieństwo, aby uzyskać koniec swojego przedziału ($0,286 + 0,286 = 0,572$)”.

Wszystkie przedziały wyznaczane są z uwzględnieniem tej reguły. Należy zwrócić uwagę, że przedziały występują zgodnie z ich posortowaniem na liście prawdopodobieństw, od największego do najmniejszego.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Aby zakodować wiadomość, kodujemy znaki po kolei.

Jeśli chcemy zakodować A, to zmniejszamy „wirtualny patyk” i zaznaczamy ponownie na nim prawdopodobieństwa. Tym razem jednak musimy się zmieścić w przedziale „zajmowanym” przez znak A, czyli od 0 do 0,286. Końce przedziałów wyznaczamy, mnożąc długość przedziału (w tym przypadku $0,286 - 0 = 0,286$) przez prawdopodobieństwo każdego ze znaków wyznaczone w kroku 7.

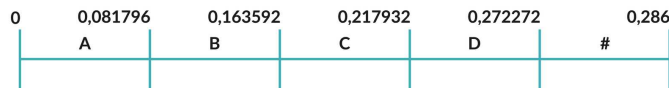
$$A = 0 + 0,286 \cdot 0,286 = 0,081796$$

$$B = 0,081796 + 0,286 \cdot 0,286 = 0,163592$$

$$C = 0,163592 + 0,286 \cdot 0,190 = 0,217932$$

$$D = 0,217932 + 0,286 \cdot 0,190 = 0,272272$$

$$\# = 0,272272 + 0,286 \cdot 0,048 = 0,286$$



13

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

14

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Zakodowanie kolejnej litery A wymagałoby dalszych podziałów odcinka odpowiadającego za literę A:

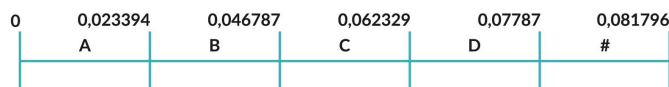
$$A = 0 + 0,081796 \cdot 0,286 = 0,023394$$

$$B = 0,023394 + 0,081796 \cdot 0,286 = 0,046787$$

$$C = 0,046787 + 0,081796 \cdot 0,190 = 0,062329$$

$$D = 0,062329 + 0,081796 \cdot 0,190 = 0,077870$$

$$\# = 0,077870 + 0,081796 \cdot 0,048 = 0,081796$$



15

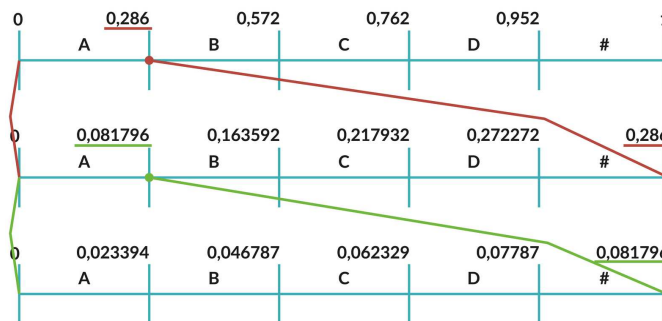
Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

16

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

W tym miejscu przestaniemy kodować tę wiadomość, ponieważ zrobienie tego do końca ręcznie zajęłoby nam zbyt dużo czasu i wymagałoby dużej liczby cyfr po przecinku. Natomiast już na tym etapie możemy zauważyć następujący schemat:



17

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

18

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Spróbujmy jednak zakodować prostszą wiadomość „BAB#” z następującymi częstościami:

- A : 1
- B : 2
- # : 1

oraz z następującymi prawdopodobieństwami:

- A : 0,25

- $B : 0,5$
- $\# : 0,25$

19

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Sortujemy je zgodnie z prawdopodobieństwami, od największego do najmniejszego, otrzymując zbiór $\{B, A, \#\}$.

Na jego podstawie tworzymy „wirtualny patyk”, który przyjmie następującą postać:

20



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

21

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Idziemy dalej zgodnie z wiadomością – zatem chcemy wejść w przedział B, tworząc kolejny podział:

22



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Następnie bierzemy kolejną literę z wiadomości, czyli A i zmniejszamy ponownie przedział.

Tym razem początkiem nowego przedziału nie jest zero, więc zrobimy to trochę wolniej:

$$Max - Min = 0,375 - 0,25 = 0,125$$

Liczbę 0,125 mnożymy przez odpowiednie prawdopodobieństwa, tzn.:

$$0,5 \cdot 0,125 = 0,0625$$

$$0,75 \cdot 0,125 = 0,09375$$

$$1 \cdot 0,125 = 0,125$$

Następnie dodajemy do 0,25 powyższe wyniki, otrzymując nowe granice przedziału:

B w granicach $< 0,25, 0,3125)$

A w granicach $< 0,3125, 0,34375)$

Hash w granicach $< 0,34375, 0,375)$

24



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Kodujemy następną literę, czyli B, zgodnie ze schematem z kodowania poprzedniej litery. Odejmujemy 0,25 od 0,3125, uzyskując długość przedziału: 0,0625.

Dzięki znajomości długości całego przedziału możemy policzyć długości przedziałów dla poszczególnych liter:

$$A = 0,0625 \cdot 0,25 = 0,015625$$

$$B = 0,0625 \cdot 0,5 = 0,03125$$

$$\# = 0,0625 \cdot 0,25 = 0,015625$$

A oto górny zakres przedziału dla poszczególnych liter:

$$B = 0,25 + 0,03125 = 0,28125$$

$$A = 0,28125 + 0,015625 = 0,296875$$

$$\# = 0,296875 + 0,015625 = 0,3125$$

Otrzymujemy ostatecznie następujący „wirtualny patyk”:



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOpIvgHK>

Kodujemy teraz ostatni symbol, czyli #. Ponieważ jest to symbol kończący, wystarczy wybrać dowolny punkt na przedziale $< 0,296875, 0,3125)$, aby zakończyć kodowanie. Wybierzemy 0,3 – ponieważ będzie

to stosunkowo łatwe do oznaczenia na „wirtualnym patyku”.

Wracamy więc do postaci wyjściowej patyka i zaznaczamy na nim punkt w miejscu 0,3.

28



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

29

Aby odczytać wiadomość, należy:

1. zapisać informację przypisaną do przedziału, w którym obecnie znajduje się zaznaczony punkt;
2. zagęścić przedział;
3. powtarzać 1 i 2, dopóki nie dotrzemy do końca dekodowania, czyli w omawianym przykładzie do znaku #.

30

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

Aby poprawnie odkodować wiadomość, musimy znać:

- udział występowania wszystkich informacji oraz jasno zdefiniowane

informacje, posortowane w kolejności, w jakiej pojawiały się na „obszarze”, który posłużył nam do kodowania;

- pozycję, w której została ostatecznie zapisana wiadomość; w omawianym przykładzie tym punktem jest 0,3;
- moment, w którym należy przestać dekodować; może o nim decydować napotkanie danego symbolu (jak w przykładzie) lub spełnienie założenia, ile informacji mamy odczytać.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

31

Odczytujemy, że postawiona przez nas kreska jest w zakresie B, wchodzimy w ten zakres, wykonując zagęszczanie, a do odkodowanej wiadomości dopisujemy B, ma ona na razie postać: „B”.

32



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

33

Następnie patrzymy na przedział, na którym mamy zaznaczony punkt, i odczytujemy, że znajduje się on w zakresie A. Wchodzimy więc

w zakres A, wykonując zagęszczanie, i do odkodowanej wiadomości dopisujemy A. Ma ona teraz postać: „BA”.

34



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

35

Odczytujemy, że zaznaczony punkt znajduje się w przedziale B, wchodzimy więc w zakres B, wykonując zagęszczenie. Do odkodowanej wiadomości dopisujemy B. Ma ona teraz postać: „BAB”.

36



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Materiał audio dostępny pod adresem:

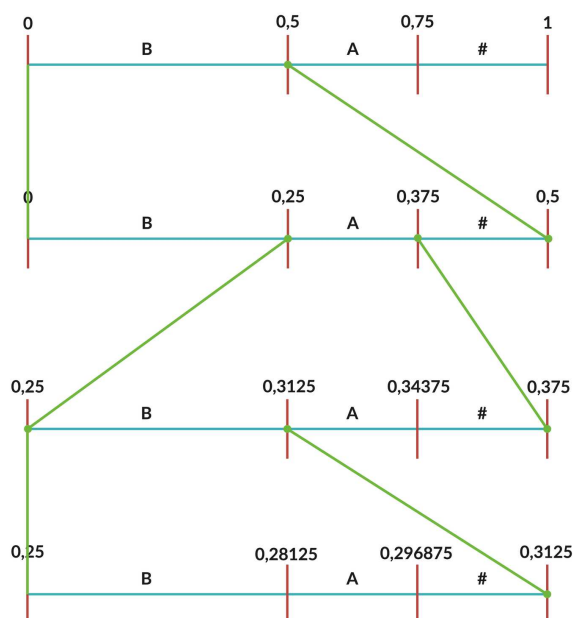
<https://zpe.gov.pl/b/PAOplvgHK>

37

Odczytujemy, że zaznaczony punkt znajduje się w zakresie #, zatem do odkodowanej wiadomości dopisujemy #. Ma ona teraz postać: „BAB#”. Ponieważ ustaliliśmy wcześniej, że znak

oznacza koniec kodowania, to w tym przypadku również kończymy odkodowywanie. Schemat kodowania prezentuje się następująco:

38



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PAOplvgHK>

39

Jak widać na przykładzie, teoretycznie istnieje możliwość zakodowania dowolnej wiadomości na patyku o długości jednego metra. W rzeczywistości w pewnej skali błędy pomiarowe i liczba nacięć mogłyby powodować pomyłki, które całkowicie zmieniałyby wiadomość. Dodatkowo w pewnym momencie szerokość ostrza stałaby się zbyt duża, aby dało się za jego pomocą kodować wiadomości.

Źródło: Contentplus.pl sp. z o.o., licencja: CC BY-SA 3.0.

Polecenie 2

Zapisz notatkę, w której zawrzesz najważniejsze informacje przedstawione w prezentacji.

Polecenie 3

Wykonaj ćwiczenia sprawdzające twoją wiedzę.

Pokaż ćwiczenia:   

Ćwiczenie 1



Przyporządkuj słowa kodowe do znaków, przy założeniu, że słowa kodowe są optymalne, a znaki mają następujące częstości: A:5, B:4, C:3, D:1. Przyjmij, że słowa kodowe zostały uzyskane z wykorzystaniem algorytmu Shannona.

Znak	Słowo Kodowe
A	
B	
C	
D	

- 1001
- 00
- 01
- 100

Ćwiczenie 2



Wskaż poprawne zdanie.

☐

Kompresja stratna to taka, która po skompresowaniu umożliwia dekompresję do postaci wejściowej.

☐

Kompresja bezstratna to taka, która nie zmienia rozmiaru pliku.

☐

Kompresja bezstratna to taka, która po skompresowaniu umożliwia dekompresję do postaci wejściowej.

☐

Kompresja stratna to taka, która nie zmienia rozmiaru pliku.

☐

Kompresja stratna to taka, która polega na utracie istotnych informacji.

Ćwiczenie 3



Zaznacz opisy kodowań odpowiednimi kolorami.

☐

Kodowanie arytmetyczne

☐

Kodowanie Shannona

☐

Kodowanie Huffmana

Charakterystyczne dla algorytmu realizującego to kodowanie jest, że w trakcie jego wykonywania tworzone jest drzewo binarne.

Charakterystyczne dla algorytmu realizującego to kodowanie jest, że w trakcie jego wykonywania logarytmujemy sumy częściowe prawdopodobieństw wystąpienia znaków.

Charakterystyczne dla algorytmu realizującego to kodowanie jest, że w trakcie jego wykonywania wchodzimy w coraz mniejsze przedziały prawdopodobieństwa i jako wynik kodowania podajemy pojedynczą liczbę.

Ćwiczenie 4



Wskaż wszystkie niepoprawne zdania.

- ☐ Kod Graya umożliwia kompresowanie wiadomości.
- ☐ Kodowanie Shannona zawsze generuje optymalne długości słów kodowych.
- ☐ Kodowanie arytmetyczne w teorii, z pominięciem praw fizyki, pozwala zakodować całą wiedzę ludzkości na dowolnie małej powierzchni.
- ☐ W kompresji bezstratnej jesteśmy w stanie uzyskać wiadomość wejściową po zdekodowaniu.

Ćwiczenie 5



Zaznacz prawdziwe zdanie.

- ☐ W kompresowaniu istotne jest, żeby czytający nie był w stanie odczytać wiadomości.
- ☐ Kodowanie zawsze powoduje kompresję.
- ☐ Kod prefiksowy stosuje się, aby można było jednoznacznie dekodować wiadomości.

Dla nauczyciela

Autor: Maurycy Gast

Przedmiot: Informatyka

Temat: Kompresja danych

Grupa docelowa:

Szkoła ponadpodstawowa, liceum ogólnokształcące, technikum, zakres rozszerzony

Podstawa programowa:

Cele kształcenia – wymagania ogólne

I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.

III. Posługiwanie się komputerem, urządzeniami cyfrowymi i sieciami komputerowymi, w tym: znajomość zasad działania urządzeń cyfrowych i sieci komputerowych oraz wykonywania obliczeń i programów.

Treści nauczania – wymagania szczegółowe

III. Posługiwanie się komputerem, urządzeniami cyfrowymi i sieciami komputerowymi.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

2) dokonuje kompresji informacji, objaśnia różnice między kompresją stratną i bezstratną tekstów, obrazów, dźwięków, filmów;

Kształtowane kompetencje kluczowe:

- kompetencje cyfrowe;
- kompetencje osobiste, społeczne i w zakresie umiejętności uczenia się;
- kompetencje matematyczne oraz kompetencje w zakresie nauk przyrodniczych, technologii i inżynierii.

Cele operacyjne (językiem ucznia):

- Wskażesz, jakie są sposoby zapisywania danych.
- Przeanalizujesz rodzaje kodowań używanych w informatyce oraz algorytmy kompresji.
- Wyjaśnisz, co to kod prefiksowy oraz czym różni się kompresja stratna od bezstratnej.

Strategie nauczania:

- konstruktywizm;
- konektywizm.

Metody i techniki nauczania:

- dyskusja;
- rozmowa nauczająca z wykorzystaniem multimediu i ćwiczeń interaktywnych;
- ćwiczenia praktyczne.

Formy pracy:

- praca indywidualna;
- praca w parach;
- praca w grupach;
- praca całego zespołu klasowego.

Środki dydaktyczne:

- komputery z głośnikami, słuchawkami i dostępem do internetu;
- zasoby multimedialne zawarte w e-materiale;
- tablica interaktywna/tablica, pisak/kreda.

Przebieg lekcji

Przed lekcją:

1. **Przygotowanie do zajęć.** Nauczyciel loguje się na platformie i udostępnia e-materiał: „Kompresja danych”. Nauczyciel prosi uczniów o zapoznanie się z treściami w sekcji „Przeczytaj”.

Faza wstępna:

1. Nauczyciel prosi uczniów o zdefiniowanie tego, czym jest kompresja. Chętna lub wybrana osoba zapisuje propozycje na tablicy.
2. Nauczyciel wyświetla temat i cele zajęć. Prosi uczniów, by na podstawie wiadomości zdobytych przed lekcją zaproponowali kryteria sukcesu.

Faza realizacyjna:

1. **Praca z tekstem.** Nauczyciel wyświetla zawartość sekcji „Przeczytaj”. Na forum klasy uczniowie analizują przedstawione w niej rozwiązania przykładów. Uczniowie metodą burzy mózgów podają przykłady kompresji.
2. **Praca z multimediu.** Nauczyciel wyświetla zawartość sekcji „Animacja”. Uczniowie dyskutują na temat sensu przeprowadzonego eksperymentu.

3. Uczniowie w parach odtwarzają eksperyment. Korzystają wyłącznie z komunikacji tekstowej.
4. Pary prezentują uzyskane rezultaty. Osoby opisujące odpowiadają na to, co było dla nich najistotniejsze do przekazania. Osoby odtwarzające opowiadają o tym, jakie para miała trudności.
5. Uczniowie wykonują ćwiczenia wskazane przez nauczyciela.

Faza podsumowująca:

1. W ramach podsumowania uczniowie dyskutują na temat znaczenia kompresji plików.
2. Nauczyciel ponownie wyświetla na tablicy temat lekcji zawarty w sekcji „Wprowadzenie” i inicjuje krótką rozmowę na temat zrealizowanych celów (czego uczniowie się nauczyli).

Praca domowa:

1. Uczniowie opisują za pomocą pseudokodu algorytm zaprezentowany w animacji.

Wskazówki metodyczne:

- Próbę odtworzenia eksperymentu studentów można poszerzyć o przygotowanie kompresji właściwych plików z wykorzystaniem algorytmu WebP.