Algorytmy i struktury danych

Lista zadań 11

Zadanie 1

Regularne drzewo binarne to takie, które nie ma węzłów z jednym dzieckiem. Udowodnij, że drzewo binarne, które nie jest regularne, nie może odpowiadać optymalnemu kodowi prefiksowemu.

Załóżmy nie wprost, że drzewo binarne, które nie jest regularne, odpowiada optymalnemu kodowi prefiksowemu. W takim drzewie istnieje węzeł, który ma tylko jedno dziecko. Węzeł ten może być usunięty, a jego dziecko może zostać podłączone do jego rodzica. W ten sposób otrzymujemy drzewo binarne, które odpowiada optymalnemu kodowi prefiksowemu, ale jest regularne. Otrzymaliśmy sprzeczność, więc założenie musi być fałszywe. Drzewo binarne, które nie jest regularne, nie może odpowiadać optymalnemu kodowi prefiksowemu.

Zadanie 2

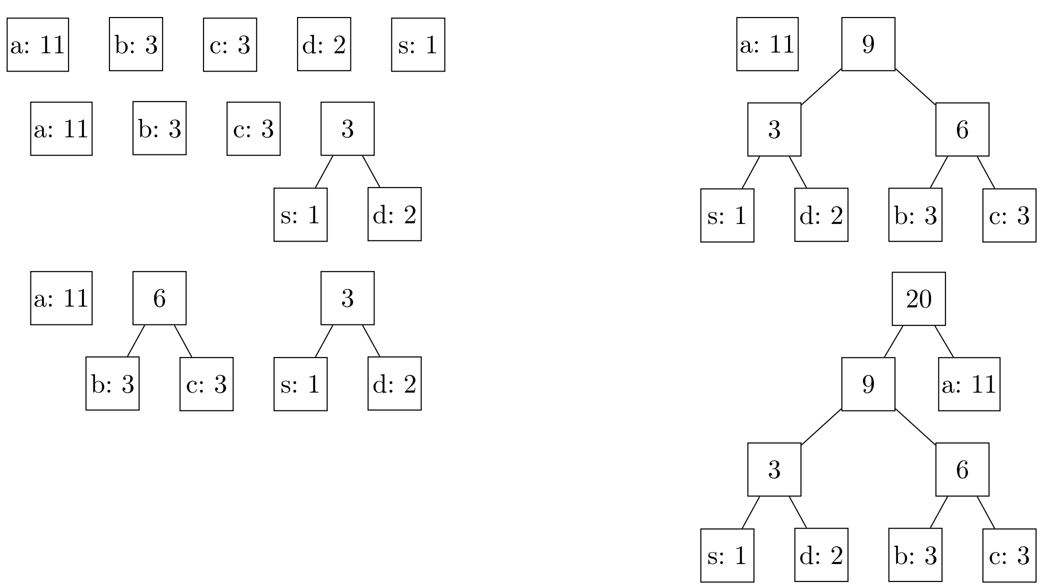
Czy kody Huffmana są wyznaczone jednoznacznie dla każdego tekstu? Dlaczego?

Tak, dla tego samego algorytmu algorytm Huffmana wyznaczy te same kody dla tego samego tekstu. Podczas tworzenia drzewa Huffmana, węzły są sortowane po częstości występowania znaku i następnie drzewa o najmniejszej częstości są scalane w jedno drzewo. Jeśli istnieją dwa drzewa o tej samej częstości, to węzły są sortowane po kolejności występowania znaku w alfabecie. W związku z tym algorytm Huffmana zawsze wyznaczy te same kody dla tego samego tekstu.

Zadanie 3

Dla podanego tekstu "bababacacadaaasadaca".

(a) Zasymuluj działanie algorytmu generującego kody Huffmana, narysuj otrzymane drzewo kodów, wypisz kody poszczególnych znaków, oraz zakodowany tekst.



a	b	c	d	s
1	010	011	001	000

b a b a b a c a c a d a a a s a d a c a
010 1 010 1 010 1 011 1 011 1 001 1 1 1 000 1 001 1 011 1

(b) Oblicz, o ile bitów otrzymana reprezentacja tekstu będzie krótsza od reprezentacji otrzymanej za pomocą kodów o stałej długości.

$$\text{Kody o stałej długości: } 20 \cdot 3 = 60$$
$$\text{Kody Huffmana: } 11 \cdot 1 + 3 \cdot 3 + 3 \cdot 3 + 2 \cdot 3 + 1 \cdot 3 = 38$$
$$60 - 38 = 22$$

(c) Mając dane drzewo kodów i zakodowany tekst wykonaj dekodowanie (zaznaczaj w ciągu bitów kreską gdzie kończą się kody poszczególnych znaków).

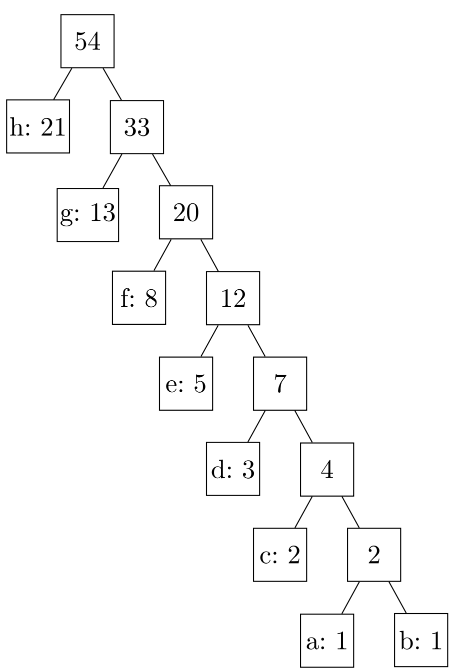
010 1 010 1 010 1 011 1 011 1 001 1 1 1 000 1 001 1 011 1
b a b a b a c a c a d a a a s a d a c a

(d) Zobacz, jak deszyfrowany tekst zmieni się, gdy zmienisz pierwszy bit zaszyfrowanej wersji na przeciwny.

1 1 010 1 010 1 010 1 1 1 011 1 001 1 1 1 000 1 001 1 011 1
a a b a b a b a a a c a d a a a s a d a c a

Zadanie 4

Jaki jest optymalny kod Huffmana, dla zbioru częstości opartego na początkowych n = 8 liczbach Fibonacciego: a:1, b:1, c:2, d:3, e:5, f:8, h:21? Uogólnij odpowiedź na przypadek dowolnego n.



Litera	Kod
a	1111110
b	1111111
c	111110
d	11110
e	1110
f	110
g	10
h	0

- Pierwsze dwa znaki mają długość $n - 1$ bitów jedynkowych, z czego pierwszy kończy się zerem.
- Długość kodu każdego kolejnego znaku jest krótsza o 1 bit od poprzedniego i ostatni bit jest zerem.

Zadanie 5

Jaki jest optymalny kod Huffmana, dla zbioru częstości opartego na liczbach: a:11, b:12, c:13, d:14, e:15, f:18, h:19? Porównaj długość zakodowanego tekstu z długością tekstu zakodowanego przy pomocy kodów stałej długości. Uogólnij odpowiedź na przypadek $n = 2^k$ liter w alfabecie, gdzie dodatkowo maksymalna ilość wystąpień jest mniejsza od dwukrotności minimalnej ilości wystąpień.

Zadanie 6

Udowodnij, że długość (ilość bitów) zaszyfrowanego tekstu, jest sumą liczb występujących w wewnętrznych węzłach drzewa kodów (nie liściach).

Zadanie 7

Narysuj na kartce sieć sortującą n liczb dla $n = 2, 4, 8, 16$. Powinna to być opisana na wykładzie sieć implementująca równoległą wersję algorytmu mergesort, działająca w czasie $O((log n)^2)$. Prześledź działanie sieci o $n = 8$ dla ciągu wejściowego: 8 4 2 3 7 5 6 1, rysując jakie liczby wchodzą i wychodzą z każdego komparatora.