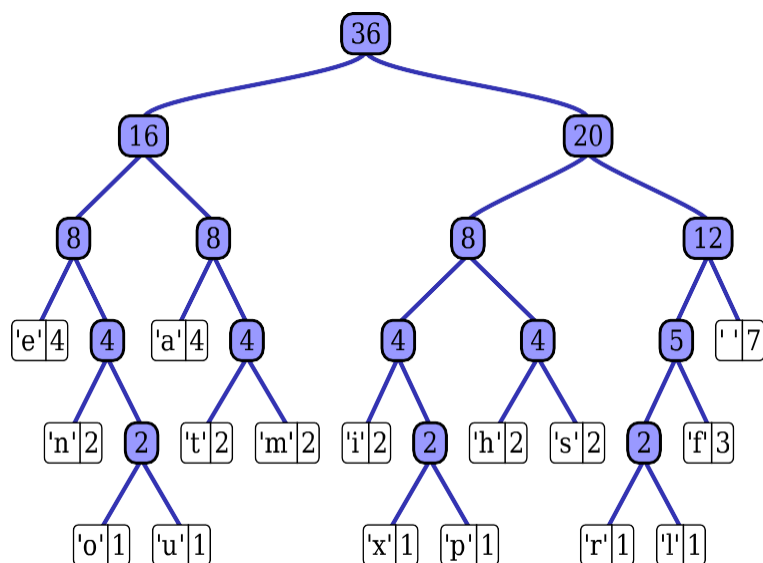


Codificare Huffman

Laboratorul 10

Codificarea Huffman (https://en.wikipedia.org/wiki/Huffman_coding) reprezintă o modalitate eficientă prin care se pot comprima și decompresa secvențe de informație. Algoritmul generează un arbore de prefixe ce pot fi folosite pentru a codifica optim informația.



Arbore Huffman pentru "this is an example of a huffman tree"

Intr-un arbore Huffman, există două tipuri de noduri:

- frunze – conțin caractere
- noduri intermediare – nu conțin informație

Pentru a construi un arbore Huffman, se execută următorii pași (mai multe detalii în pagina de pe Wikipedia):

1. Se calculează frecvența de apariție a fiecărui caracter din text
2. Se creează câte un nod frunză pentru fiecare caracter cu frecvență diferită de 0
3. Se introduc toate nodurile într-un minheap cu prioritățile = frecvențele calculate la pasul 1
4. Cât timp mai sunt minim două noduri în minheap:
 - a. se extrag primele două noduri (removeMin)
 - b. se creează un nod nou fără valoare care este părintele celor 2 noduri de la a
 - c. se inserează noul nod în heap cu prioritatea = suma priorităților copiilor
5. Ultimul nod rămas în minheap este rădăcina arborelui Huffman

Având arborele, el este interpretat astfel: coborârea “la stânga” este “0”, coborârea “la dreapta” este “1”. Astfel, pe exemplul de mai sus, 'o' se codifică '00110'. Având această informație, orice text poate fi codificat și decodificat.

Aveti de implementat urmatoarele functii:

- **PHuffmanNode initNode(unsigned char value)** – creeaza un nod nou cu valoarea **value**. Fiecare frecventa va fi salvata pe pozitia egala cu valoarea caracterului in tabela ascii.
- **void computeFreqs(char *text, int freqs[256])** - numara frecventele de aparitie ale fiecarui caracter din **text** si salveaza frecventele in vectorul **freqs**
- **PHuffmanNode makeTree(int freqs[256])** - creeaza arborele Huffman pentru vectorul de frecvente dat ca parametru. Functia va intoarce radacina arborelui Huffman. Va veti folosi de minheap-ul din arhiva pentru a implementa algoritmul Huffman. Valoarea unui nod in arborele Huffman este nodul de tip PHuffmanNode. Campul prior va fi egal cu frecventa de aparitie a nodului.
- **void makeCodes(PHuffmanNode root, char **allCodes)** – genereaza codurile pentru toate caracterele din arborele **root**. Parametrul allCodes este un vector de stringuri unde vor fi salvate codurile. Fiecare cod va fi salvat pe pozitia data de caracter in tabela ascii. Vectorul nu are memorie alocata pentru stringuri.
- **char *compress(char *textToEncode, char **allCodes)** – intoarce codificarea asociata textului **textToEncode** pe baza codurilor din **allCodes**
- **char *decompress(char *textToDecode, PHuffmanNode root)** – intoarce textul decodificat. Primeste radacina arborelui **root** si textul **textToDecode**