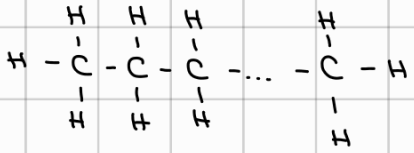
Arbori

① $C_{31} H_{64}$ - e arbore



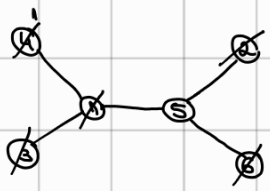
$n = 95$

$m = n - 1 = 94$

fără cicluri

conex

$C_m H_{2m+2}$

② 

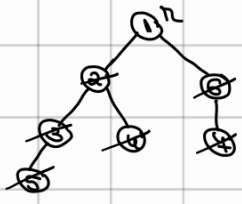
$n = 5$

$V = \{1 \dots n\}$

Prüfer (T)

1. $K = 0$
2. while T contine mai mult de rădăcină
3. fie v frunza minimă
4. $K \leftarrow$ părinte(v)
5. $T = T \setminus \{v\}$
6. return K

$n = 5$ K: 5, 1, 1, 5, 5



K: 2, 3, 2, 1, 6, 1

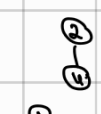
2 3 2 1 6 1

DECODARE - PRÜFER (K, n)

1. $T = \emptyset$
2. for $i = 1$ to $n-1$ do
3. x primul el din dim k
4. y primul nr nat (crescător) ce nu e în K
5. $(x, y) \in E(T)$, x părinte y
6. șterg x din K, adaug y în coada lui K
7. return T

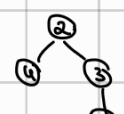
x						
2	3	2	1	6	1	4
3	2	1	6	1	4	5
2	1	6	1	4	5	3
1	6	1	4	5	3	2
6	1	4	5	3	2	7
1	4	5	3	2	7	6

i = 1)

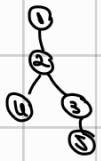


i = 2)

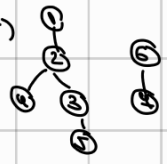
i = 3)



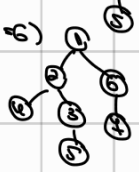
i = 4)



i = 5)



i = 6)



000 101 101 101 100 111 1101 1100 1100   29 biți
aaa bbb c d e f f

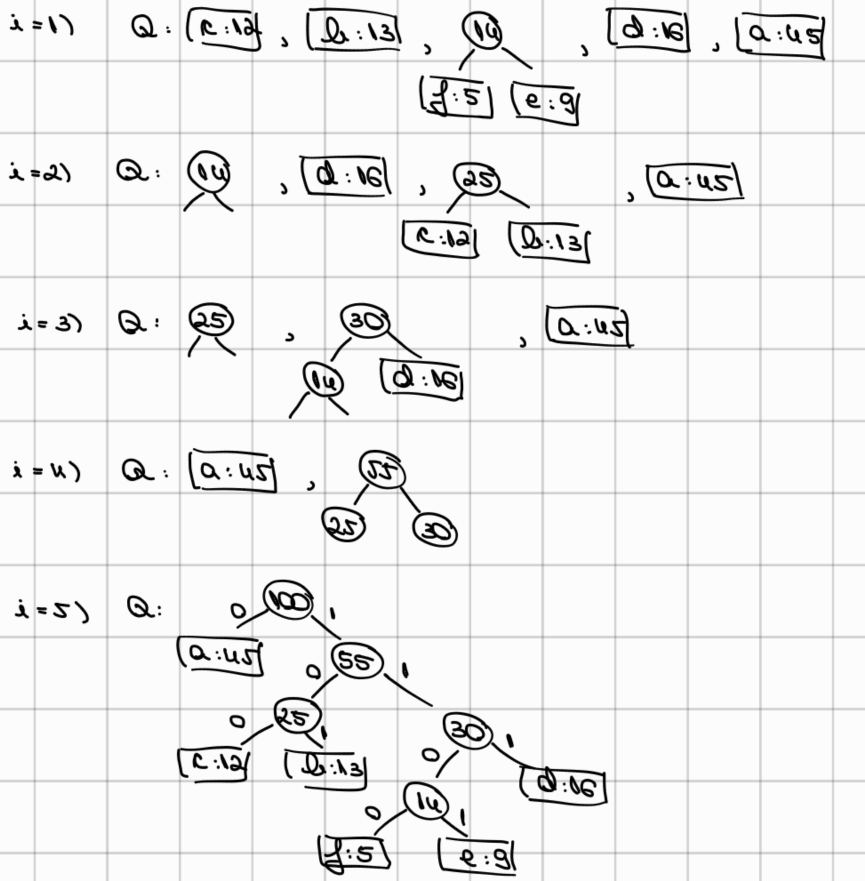
ASCII   11 caractere → 88 biți

Huffman (C)

```
1. n = |C|  (nr caractere)
2. Q = C
3. for i = 1 to n-1 do
4.     alocă un nod nou, z
5.     z stâng = x = EXTRACT_MIN(Q)
6.     z drept = y =  — " —
7.     z.f = x.f + y.f
8      INSERT(Q, z)
9. return EXTRACT_MIN(Q)
```

	a	b	c	d	e	f
f	45	13	12	16	9	5

Q: f:5, e:9, c:12, b:13, d:16, a:45

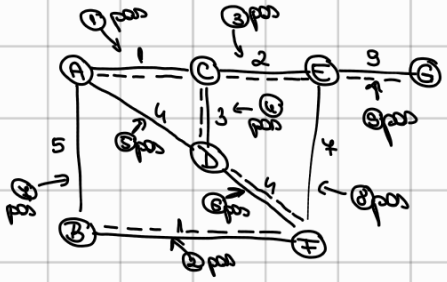


a: 0
b: 101
c: 100
d: 111
e: 1101
f: 1100

NICIUN cod nu e PREFIX pt alt cod

Kruskal (G, w)

1. A = ∅
2. for v ∈ V do
3. MAKE-SET(v)
4. sortare crescător muchii după w (după pondere)
5. for (u,v) ∈ E luate crescător după w
6. if FIND-SET(u) ≠ FIND-SET(v)
7. A = A ∪ {(u,v)}
8. UNION(u,v)
9. return A



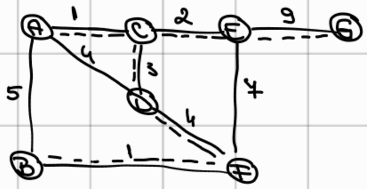
A: {(A,C), (B,F), (C,E), (C,D), (D,F), (E,G)

~~{A}~~, ~~{B}~~, ~~{C}~~, {D}
{E}, ~~{F}~~, {G}
~~{A,C}~~, {B,F}
~~{A,E,C}~~
~~{A,E,C,D}~~
~~{A,B,E,C,D,E,F}~~
{A ... G}

PRIM (G, w, r)

1. for v ∈ V do
2. v.key = ∞
3. v.π = NIL
4. r.key = 0
5. Q = V
6. while Q ≠ ∅ do
7. u = EXTRACT-MIN(Q)
8. for v ∈ G.Adj[u] (v - vecin lui u)
9. if v ∈ Q și w(u,v) < v.key
10. v.key = w(u,v)
11. v.π = u

(parimble)



r = F

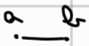
v ∈ V	A	B	C	D	E	F	G
v.key	~~∞~~ ~~5~~ ~~4~~ 1	~~∞~~ 1	~~∞~~ 3	~~∞~~ 4	~~∞~~ ~~7~~ 2	0	~~∞~~ 9
v.π	~~/~~ ~~B~~ ~~D~~ C	~~/~~ F	~~/~~ D	~~/~~ F	~~/~~ ~~F~~ C	–	~~/~~ F

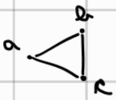
Q: ~~F~~ 0, ~~A~~ ~~∞~~ ~~5~~ ~~4~~ 1, ~~B~~ ~~∞~~ 1, ~~C~~ ~~∞~~ 3, ~~D~~ ~~∞~~ 4, ~~E~~ ~~∞~~ ~~7~~ 2, G ∞

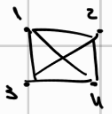
G = (V, E), neponderat

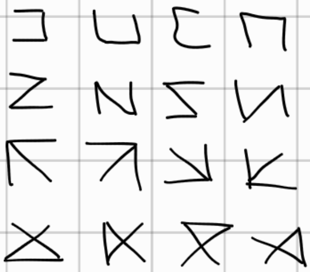
V = {1, ..., n} |T| = ? = n^{n-2}

arbori acoperire

n = 2 

n = 3 

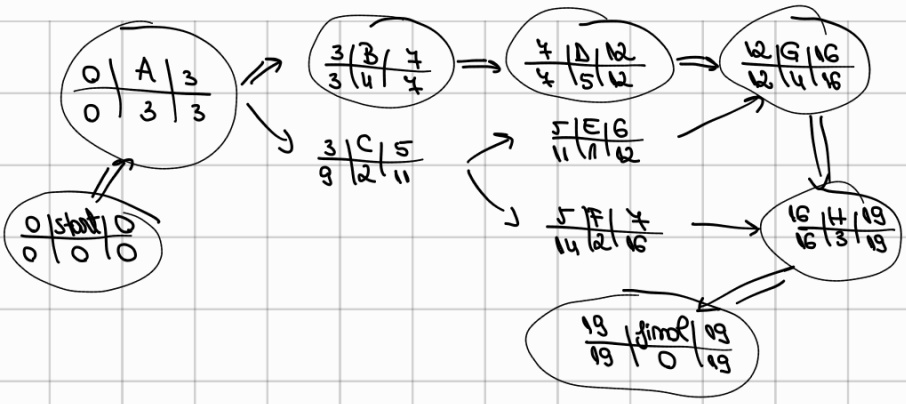
n = 4 



Drum critic

activitate	predecesori	durata
A	–	3
B	A	4
C	A	2
D	B	5
E	C	1
F	C	2
G	D, E	4
H	F, G	3

ES	act	EF
LS	durata	LF


0 A 3
0 3 3
3 B 7
3 4 7
7 D 12
7 5 12
12 G 16
12 4 16
3 C 5
3 2 11
11 E 16
11 1 12
11 F 7
14 2 16
16 H 19
16 3 19
19 final 19
19 0 19
0 start 0
0 0 0