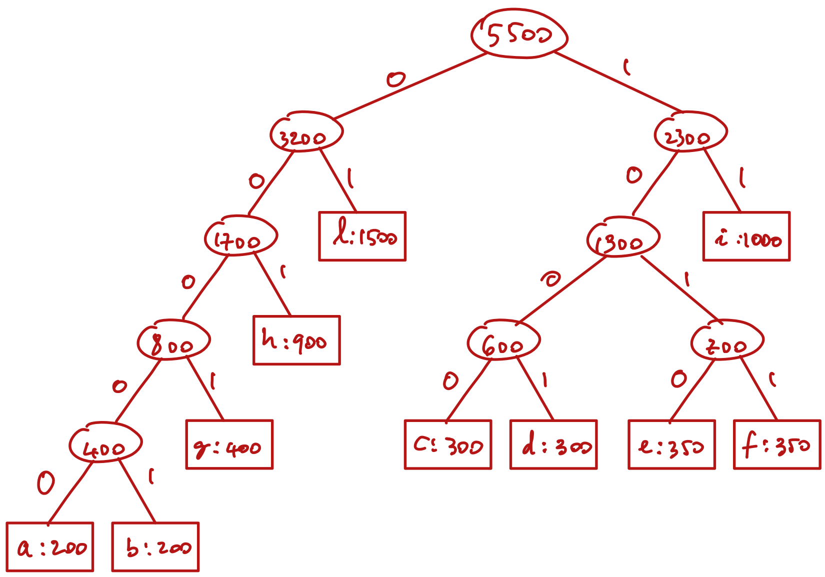
ESERCIZIO 2 (Foglio A)

Sia T un testo di 5500 caratteri in un alfabeto con i simboli $a, b, c, d, e, f, g, h, i, \ell$, le cui frequenze siano rispettivamente 200, 200, 300, 300, 350, 350, 400, 900, 1000, 1500.
Dopo aver definito la nozione di *codice prefisso*, si determini il numero minimo di bit necessari per rappresentare il testo T utilizzando un codice prefisso ottimo, illustrando l'algoritmo utilizzato (anche mediante pseudo-codice).
Qual è il risparmio percentuale rispetto ad una codifica minimale di T con un codice a lunghezza fissa?



$(2\cdot 200)\cdot 5 = 2000$

$(400+600+700)\cdot 4 = 6800$

$900 \cdot 3 = 2700$

$(1500+1000)\cdot 2 = 5000$

16500 bit con codifica di Huffman

cod(b) = 00001

cod(d) = 1001

⋮

per la codifica a lunghezza fissa sono necessari e sufficienti 4 bit x carattere

$4 \times 5500 = 22000$ bit con codifica a lunghezza fissa

\# bit risparmiati = $22000 - 16500 = 5500$ bit

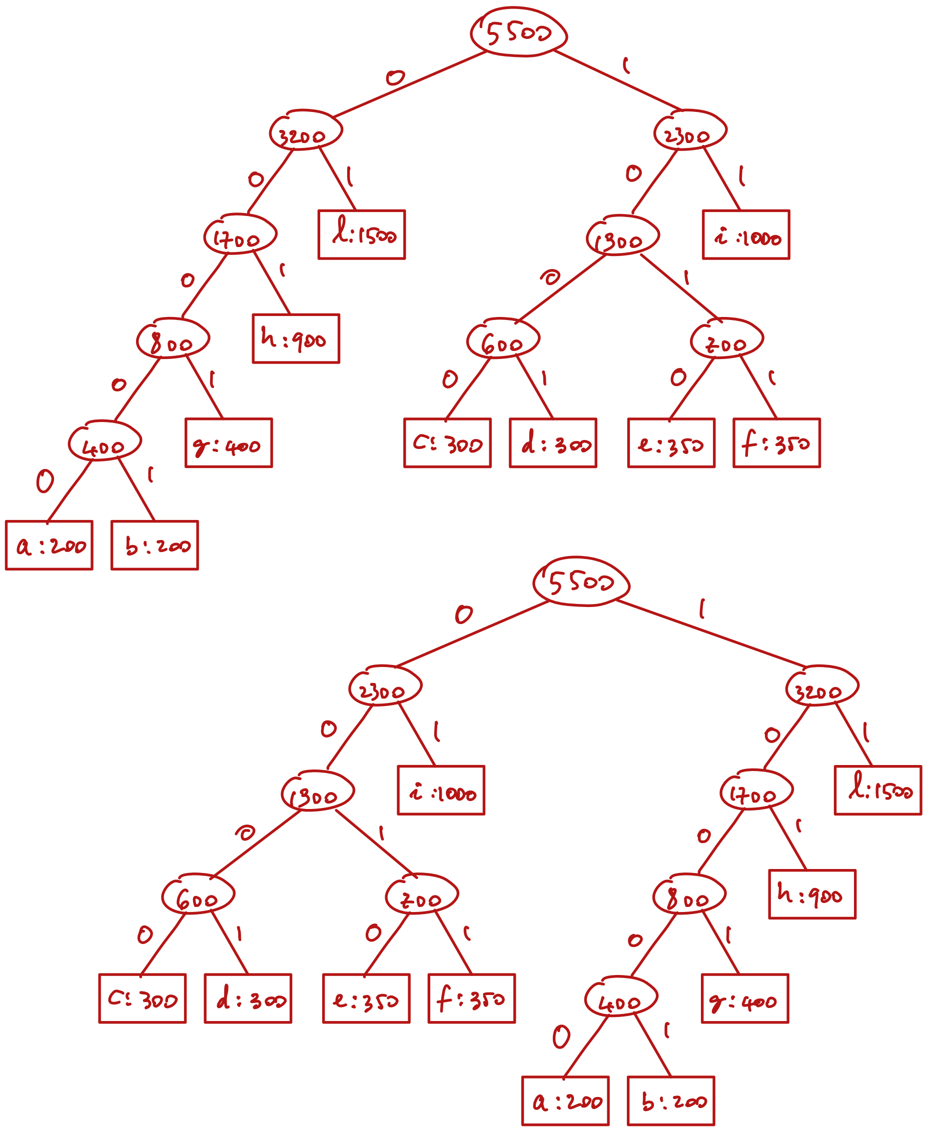
risparmio percentuale = 25%

$22000 : 100 = 5500 : x$

$$x = \frac{100 \cdot 5500}{22000} = 25\ \%$$

UN'OSSERVAZIONE

Gli alberi di decodifica ...



... sono equivalenti!

ESERCIZIO 2

(A) Data la funzione $h(x,i) =_{Def} (x+3i) \mod 17$, si illustri l'inserimento delle chiavi

$$23, 43, 21, 5, 62, 72, 58, 48, 52, 46, 78, 55, 35, 17, 51$$

in una tabella hash di dimensione 17, inizialmente vuota e organizzata con l'indirizzamento aperto, utilizzando $h(x,i)$ come funzione hash.

(B) Si enunci l'ipotesi di *hashing uniforme*, si forniscano dei limiti superiori al numero medio di scansioni in ricerche *con* e *senza* successo in una tabella hash con fattore di carico α, assumendo l'ipotesi di hashing uniforme.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17	52	35	51	21	5	23	72		43	58	62	46	78	48		55

$h(23,0) = 6$

$h(43,0) = 9$

$h(21,0) = 4$

$h(5,0) = 5$

$h(62,0) = 11$

$h(72,0) = 4$, $h(72,1) = (72+3) \mod 17 = 75 \mod 17 = 7$

$= ((72 \mod 17) + (3 \mod 17)) \mod 17$

$= ((72 \mod 17) + 3) \mod 17$

$= (4+3) \mod 17 = 7$

$h(58,0) = 7 \longrightarrow 10$

$h(48,0) = 14$

$h(52,0) = 1$

$h(46,0) = 12$

$h(78,0) = 10 \longrightarrow 13$

$h(55,0) = 4 \longrightarrow 7 \longrightarrow 10 \longrightarrow 13 \longrightarrow 16$

$h(35,0) = 1 \longrightarrow 4 \longrightarrow 7 \longrightarrow 10 \longrightarrow 13 \longrightarrow 16 \longrightarrow 2$

$h(17,0) = 0$

$h(51,0) = 0 \longrightarrow 3$

ESERCIZIO 2

Si descriva l'algoritmo Counting Sort (campo di applicazione, pseudocodice, complessità, proprietà, ecc.) e lo si illustri sull'array di coppie

$$A = [(8, A), (0, B), (7, C), (9, D), (8, E), (0, F), (7, G), (9, H), (8, I), (0, L)]$$

da ordinare rispetto alla prima componente.

	0	1	2	3	4	5	6	7	8	9
C =	3	0	0	0	0	0	0	2	3	2

	0	1	2	3	4	5	6	7	8	9
C =	0	3	3	3	3	3	3	3	5	8

	1	2	3	4	5	6	7	8	9	10
B =	(0,B)	(0,F)	(0,L)	(7,C)	(7,G)	(8,A)	(8,E)	(8,I)	(9,D)	(9,H)

ESERCIZIO 2 **(Foglio B)**

Sia $\otimes$ un'operazione *associativa* su matrici di numeri reali tale che, date due matrici A e B rispettivamente di dimensioni $p \times q$ e $q \times r$, produce una matrice $A \otimes B$ di dimensione $p \times r$, effettuando $pq + qr$ operazioni elementari.
Sia $\mathcal{A} = (A_1, A_2, \ldots, A_n)$ una sequenza di matrici di dimensioni $p_{i-1} \times p_i$, per $i = 1, 2, \ldots, n$.
Utilizzando la metodologia della programmazione dinamica, si descriva un'algoritmo per determinare la parentesizzazione della sequenza $\mathcal{A}$ che consenta di calcolare la matrice

$$A_1 \otimes A_2 \otimes \ldots \otimes A_n$$

con il minor numero possibile di operazioni elementari.
Qual è la complessità dell'algoritmo trovato in funzione della lunghezza n della sequenza $\mathcal{A}$?

$$(A_i \otimes \ldots \otimes A_k) \otimes (A_{k+1} \otimes \ldots \otimes A_j)$$

$$p_{i-1} \times p_k \qquad p_k \times p_j$$

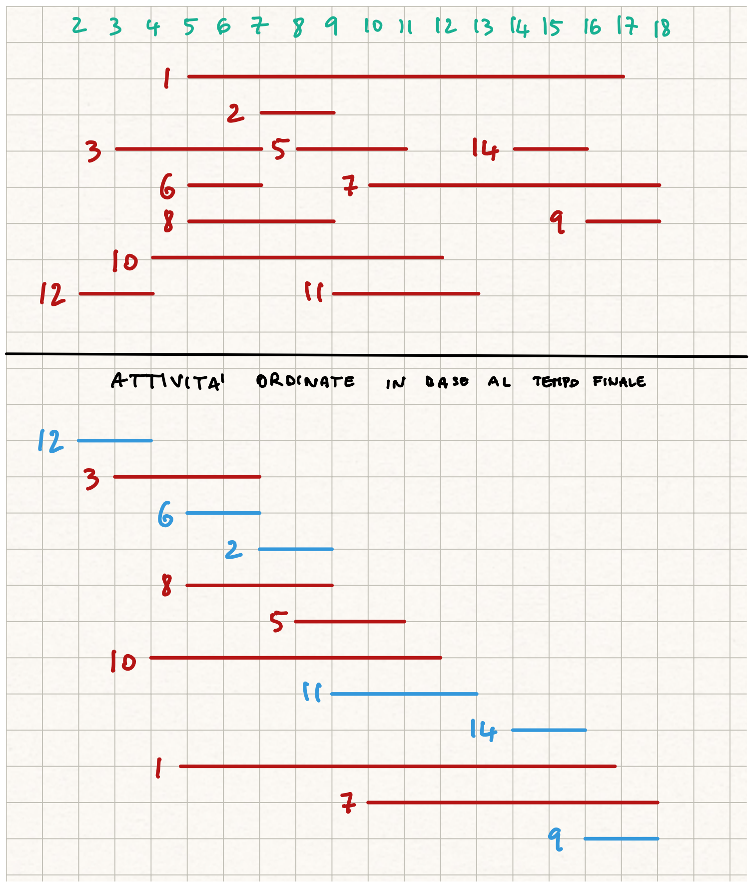
$$m[i,j] = \begin{cases} 0 & \text{se } i = j \\ \min\limits_{i \le k < j} \left(m[i,k] + m[k+1,j] + p_{i-1}p_k + p_k p_j \right) & \text{se } i < j \end{cases}$$

$p_{i-1}p_kp_j$

ESERCIZIO 2

Nel contesto della metodologia *greedy*, si enunci il problema di ottimizzazione relativo alla *selezione di attività* e se ne discuta una soluzione efficiente, anche mediante pseudo-codice, valutandone la complessità computazionale e illustrandola sull'insieme $S = \{a_1, \ldots, a_{12}\}$ di 12 attività caratterizzate dai seguenti tempi iniziali e finali:

i	1	2	3	4	5	6	7	8	9	10	11	12
s_i	5	7	3	14	8	5	10	5	16	4	9	2
f_i	17	9	7	16	11	7	18	9	18	12	13	4



$$\boxed{A = \{12, 6, 2, 11, 14, 9\}}$$

ESERCIZIO 4

Si supponga di eseguire l'algoritmo per il calcolo della distanza di editing, basato sulla tecnica di programmazione dinamica, assumendo che le stringhe in input siano X = "$GTAAGT$" e Y = "$GCTAAGA$". Fornire la configurazione finale della matrice ottenuta durante l'esecuzione dell'algoritmo.

$$M[i,j] = \begin{cases} j & \text{SE } i = 0 \\ i & \text{SE } j = 0 \\ \min \{ M[i,j-1]+1, \\ \quad M[i-1,j]+1, \\ \quad M[i-1,j-1] + p(i,j) \} & \text{SE } i \geq 1 \text{ E } j \geq 1 \end{cases}$$

i \ j		0	1	2	3	4	5	6	7
			G	C	T	A	A	G	A
0		0	1	2	3	4	5	6	7
1	G	1	0	1	2	3	4	5	6
2	T	2	1	1	1	2	3	4	5
3	A	3	2	2	2	1	2	3	4
4	A	4	3	3	3	2	1	2	3
5	G	5	4	4	4	3	2	1	2
6	T	6	5	5	4	4	3	2	2

DISTANZA-EDIT (GTAAGT, GCTAAGA) = 2

ALLINEAMENTO DI COSTO 2 = M I M M M M S

```
G - T A A G T
M I M M M M S
G C T A A G A
```

ESERCIZIO 4

Si supponga di eseguire l'algoritmo per il calcolo della lunghezza della LCS, basato sulla tecnica di programmazione dinamica, assumendo che le stringhe in input siano X = "ACCACBCA" e Y = "CBABBACA".
Fornire la configurazione finale della matrice ottenuta durante l'esecuzione dell'algoritmo.

$$c[i,j] = \begin{cases} 0 & \text{SE } i=0 \text{ O } j=0 \\ c[i-1,j-1]+1 & \text{SE } i,j>0 \text{ E } x_i = y_j \\ \max(c[i,j-1], c[i-1,j]) & \text{SE } i,j>0 \text{ E } x_i \neq y_j \end{cases}$$

i \ j		0	1	2	3	4	5	6	7	8
			C	B	A	B	B	A	C	A
0		0	0	0	0	0	0	0	0	0
1	A	0	0	0	1	1	1	1	1	1
2	C	0	1	1	1	1	1	1	2	2
3	C	0	1	1	1	1	1	1	2	2
4	A	0	1	1	2	2	2	2	2	3
5	C	0	1	1	2	2	2	2	3	3
6	B	0	1	2	2	3	3	3	3	3
7	C	0	1	2	2	3	3	3	4	4
8	A	0	1	2	3	3	3	4	4	5

A C C - A C B - C A

- - C B A B B A C A

A C C A C B - C A

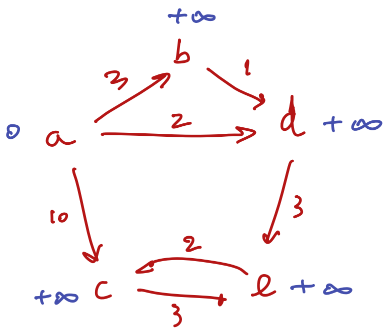
- C B A B B A C A

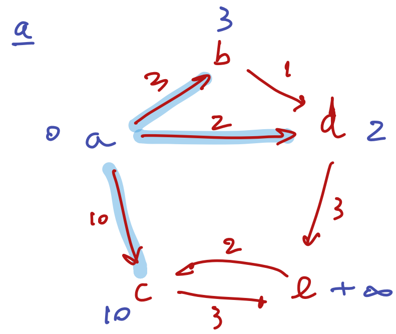
ESERCIZIO 5

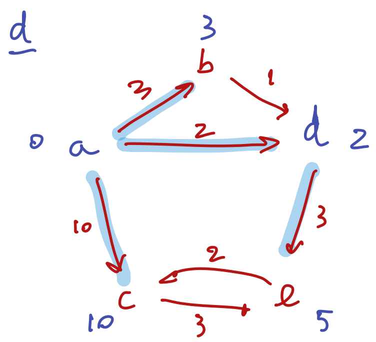
Sia dato il grafo orientato e pesato $G = (V, E)$ dove:

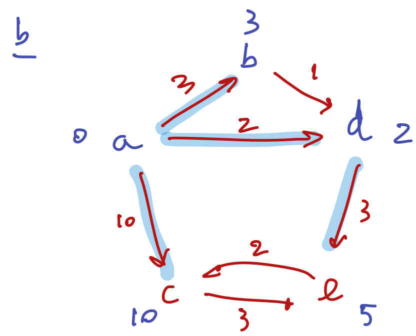
$$V = \{a, b, c, d, e\}, \qquad E = \{(a, b, 3), (a, d, 2), (b, d, 1), (a, c, 10), (c, e, 3), (d, a, 2), (e, c, 2), (d, e, 3)\}.$$

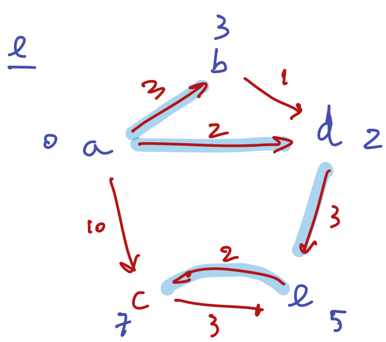
Si supponga di eseguire l'algoritmo di Dijkstra sul grafo G con nodo sorgente a. Indicare (disegnandola graficamente) la configurazione del grafo dopo ogni iterazione dell'algoritmo, specificando la stima di cammino minimo di ciascun nodo del grafo.

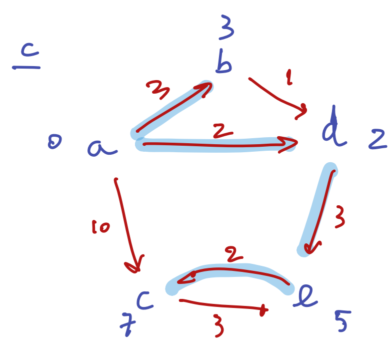












ESERCIZIO 3

Sia data la sequenza di 6 matrici definito dal vettore delle dimensioni $\langle 4, 2, 2, 5, 3, 3, 2\rangle$ e si supponga di eseguire l'algoritmo di programmazione dinamica per il calcolo della parentesizzazione ottima di una sequenza di matrici.

Si fornisca il costo ottimale della moltiplicazione della sequenza fornita in input, la matrice ottenuta alla fine dell'esecuzione dell'algoritmo e la relativa parentesizzazione ottima.

$$m[i,j] = \begin{cases} 0 & i=j \\ \min\limits_{i \le k < j} \left(m[i,k] + m[k+1,j] + p_{i-1} p_k p_j \right) & i<j \end{cases}$$

m

		2	2	5	3	3	2
	i \ j	1	2	3	4	5	6
4	1	0	16	56	66	84	92
2	2	–	0	20	42	60	68
2	3	–	–	0	30	48	60
5	4	–	–	–	0	45	48
3	5	–	–	–	–	0	18
3	6	–	–	–	–	–	0

s

i \ j	1	2	3	4	5	6
1	–	1	2	1	1	1
2	–	–	2	2	2/4	2
3	–	–	–	3	4	4/5
4	–	–	–	–	4	4
5	–	–	–	–	–	5
6	–	–	–	–	–	–

$m[1,2] = 0 + 0 + 4 \cdot 2 \cdot 2 = 16$ $\qquad s[1,2] = 1$

$m[2,3] = 0 + 0 + 2 \cdot 2 \cdot 5 = 20$ $\qquad s[2,3] = 2$

$m[3,4] = 0 + 0 + 2 \cdot 5 \cdot 3 = 30$ $\qquad s[3,4] = 3$

$m[4,5] = 0 + 0 + 5 \cdot 3 \cdot 3 = 45$ $\qquad s[4,5] = 4$

$m[5,6] = 0 + 0 + 3 \cdot 3 \cdot 2 = 18$ $\qquad s[5,6] = 5$

$$m[1,3] = \min(0 + 20 + 4 \cdot 2 \cdot 5,\ 16 + 0 + 4 \cdot 2 \cdot 5) = \min(60, 56) = 56$$

$s[1,3] = 2$

$$m[2,4] = \min(0 + 30 + 2 \cdot 2 \cdot 3,\ 20 + 0 + 2 \cdot 5 \cdot 3) = \min(42, 50) = 42$$

$s[2,4] = 2$

$$m[3,5] = \min(0+45+2\cdot5\cdot3,\ 30+0+2\cdot3\cdot3) = \min(75, 48) = 48 \qquad s[3,5] = 4$$

$$m[4,6] = \min(0+18+5\cdot3\cdot2,\ 45+0+5\cdot3\cdot2) = \min(48, 75) = 48 \qquad s[4,6] = 4$$

$$m[1,4] = \min(0+42+4\cdot2\cdot3,\ 16+30+4\cdot2\cdot3,\ 56+0+4\cdot5\cdot3) = \min(66, 70, 116) = 66 \qquad s[1,4] = 1$$

$$m[2,5] = \min(0+48+2\cdot2\cdot3,\ 20+45+2\cdot5\cdot3,\ 42+0+2\cdot3\cdot3) = \min(60, 95, 60) = 60 \qquad s[2,5] = 2/4$$

$$m[3,6] = \min(0+48+2\cdot5\cdot2,\ 30+18+2\cdot3\cdot2,\ 48+0+2\cdot3\cdot2) = \min(68, 60, 60) = 60 \qquad s[3,6] = 4/5$$

$$m[1,5] = \min(0+60+4\cdot2\cdot3,\ 16+48+4\cdot2\cdot3,\ 56+45+4\cdot5\cdot3,\ 66+0+4\cdot3\cdot3) = 84 \qquad s[1,5] = 1$$

$$m[2,6] = \min(0+60+2\cdot2\cdot2,\ 20+48+2\cdot5\cdot2,\ 42+18+2\cdot3\cdot2,\ 60+0+2\cdot3\cdot2) = 68 \qquad s[2,6] = 2$$

$$m[1,6] = \min\begin{pmatrix} 0+68+4\cdot2\cdot2, \\ 16+60+4\cdot2\cdot2, \\ 56+48+4\cdot5\cdot2, \\ 66+18+4\cdot3\cdot2, \\ 84+0+4\cdot3\cdot2 \end{pmatrix} = 84 \qquad s[1,6]=1$$

COSTRUZIONE DELLE PARENTESIZZAZIONI OTTIME

s

i \ j	1	2	3	4	5	6
1	–	1	2	1	1	1
2	–	–	2	2	2/4	2
3	–	–	–	3	4	4/5
4	–	–	–	–	4	4
5	–	–	–	–	–	5
6	–	–	–	–	–	–

$\langle 4, 2, 2, 5, 3, 3, 2\rangle$

4x2 2x2 2x5 5x3 3x3 3x2

$$\boxed{A_1 \times (A_2 \times ((A_3 \times A_4) \times (A_5 \times A_6)))}$$

VERIFICA $4\cdot2\cdot2 + 2\cdot2\cdot2 + 2\cdot5\cdot3 + 2\cdot3\cdot2 + 3\cdot3\cdot2$

$= 16+8+30+12+18 = 24+30+30 = 84$

4x2 2x2 2x5 5x3 3x3 3x2

$$\boxed{A_1 \times (A_2 \times (((A_3 \times A_4) \times A_5) \times A_6))}$$

VERIFICA $2\cdot5\cdot3 + 2\cdot3\cdot3 + 2\cdot3\cdot2 + 2\cdot2\cdot2 + 4\cdot2\cdot2 = 84$