ESERCIZIO 2

$s_0 = 1$

$l_0 = 0$

$p(a) = 0,2 = \frac{1}{5}$

$p(b) = p(c) = 0,4 = \frac{2}{5}$

ABC

$f(a) = 0$

$f(b) = \frac{1}{5}$

$f(c) = \frac{3}{5}$

① $s_1 = 1 \cdot \frac{1}{5} = \frac{1}{5}$

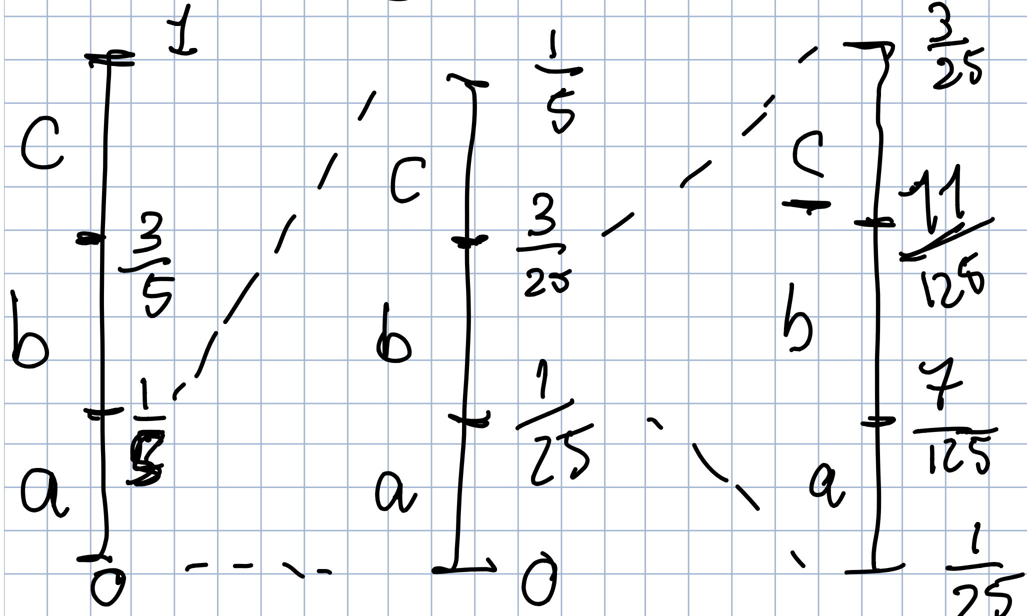
$l_1 = 0 + 1 \cdot 0 = 0$

② $s_2 = \frac{1}{5} \cdot \frac{2}{5} = \frac{2}{25}$

$l_2 = 0 + \frac{1}{5} \cdot \frac{1}{5} = \frac{1}{25}$

③ $s_3 = \frac{2}{25} \cdot \frac{2}{5} = \frac{4}{125}$

$l_3 = \frac{1}{25} + \frac{2}{25} \cdot \frac{3}{5} = \frac{6}{125} + \frac{1}{25} = \frac{11}{125}$



$$\left(x \in \left[\frac{11}{125}, \frac{15}{125}\right), 3\right)$$

$$\frac{11}{125} + \frac{4}{125} \cdot \frac{1}{2} = \frac{22+4}{250} = \frac{26}{250}$$

$$\log_2 \frac{2}{h} = \log_2 \frac{500}{26} = \log_2 19 = 5 \text{ bit}$$

$\frac{26}{250} \cdot 2 < 1$ si → output 1

$\frac{52}{250} \cdot 2 < 1$ si → output 1

$\frac{104}{250} \cdot 2 < 1$ si → output 1

$\frac{208}{250} \cdot 2 < 1$ NO → output 0

$$\frac{416}{250} - 1 = \frac{416}{250} - \frac{250}{250} = \frac{166}{250}$$

$166 \cdot 2 < 1$ NO → output 0

$\overline{250}$

$$\boxed{1\ 1\ 1\ 0\ 0\ 1}$$

ESERCIZIO 4

(3) Arithmetic Coding is 2 bit from the optimal lower bound given by the entropy of a string/source.

Dim: Data una stringa S a n sappiamo che la quantità di bit che viene emessa è

$$\lg_2 \frac{2}{S_n} < 2 - \lg S_n = 2 - \lg_2 \prod_{i=1} P[S[i]]$$

$$= 2 - \sum_{i=1}^{n} \lg_2 P[S[i]]$$

Ora data la stringa lunga n, $\forall$ simbolo che compare nella stringa calcoliamo

$$P[\sigma] = \frac{n_\sigma}{n}$$

Quindi:

$$2 - \sum_{\sigma \in \Sigma} n_\sigma \log_2 P[\sigma] = 2 - n\left(\sum_{\sigma \in \Sigma} P[\sigma] \log_2 P[\sigma]\right)$$

$$2 + n H_0$$

ESERCIZIO 1

```
Reservoir Sampling(S):
    Crea R[1,m] = S[1,m]
    j = m+1
    for each next item in S
        h = rand(1, j)
        if (h ≤ m)
            set R[h] = S[m]
            j++
    Return R
```

Simulazione algoritmo:

$n = 8$ $[a, b, c, d, e, f, g, h]$ $m = 2$

random integer: $[3, 1, 2, 2, 1, 5]$

$R = [a, b]$

$h=3 \rightarrow$ No

$h=1 \rightarrow R[1]=d \rightarrow R[d,b]$

$h=2 \rightarrow R[2]=e \rightarrow R[d,e]$

$h=2 \rightarrow R[2]=f \rightarrow R[d,f]$

$h=1 \rightarrow R[1]=g \rightarrow R[g,f]$

$h=5 \rightarrow$ niente

alla fine il sample è $[g,f]$

ESERCIZIO 4

(1) BoundedQS (S, i, j)

while $(j-i \geq n_0)$:

r = pivot random in S
swap S[i] con S[r]
p = partition (S, i, j)

if $\left(p \leq \frac{i+j}{2}\right)$

BoundedQS $(S, i, p-1)$

$i = p+1$

else if $(p \geq \frac{i+j}{2})$

BoundedQS$(S, p+1, j)$

$j = p-1$

InsertionSort(S, i, j)

Con il classico QuickSort abbiamo un costo in spazio $O(n)$ dovuto alle chiamate ricorsive. Con il bounded QS riduciamo questo costo a $O(\lg n)$ perché riduciamo le chiamate ricorsive. Infatti si usa la ~~ta~~ elimination of tail recursion e andiamo ogni volta a fare la ricorsione sulla parte più piccola dell'array, sulla più grande invece andiamo avanti con il while poi la dividiamo e di nuovo richiamiamo BoundedQS sulla parte più piccola

ESERCIZIO 3

$\{aa, ad, bc, bd, ca, dc\}$

$rank(x) = 3, 4, 5, 6$ $\qquad m = 13$

$\quad\quad\quad\quad a \;\; b \;\; c \;\; d$

$h_1(x' x'') = 3 \cdot rank(x') \cdot rank(x'') \bmod 13$

$h_2(x', x'') = rank(x') + rank(x'') \bmod 13$

Design a proper $g(t)$

$a = 3 \qquad c = 5$

$b = 4 \qquad d = 6$

$h_1(x' x'') = 3 \cdot rank(x') \cdot rank(x'') \bmod 13$

$h_2(x', x'') = rank(x') + rank(x'') \bmod 13$

<u>aa</u> : $h_1 = 3 \cdot 3 \cdot 3 \bmod 13 = 1$

$h_2 = 3 + 3 \bmod 13 = 6$

$\underline{ad}$ = h_1: $3 \cdot 3 \cdot 6 \mod 13 = 2$

$h_2 = 3 + 6 \mod 13 = 9$

$\underline{bc}$ h_1: $3 \cdot 4 \cdot 5 \mod 13 = 8$

h_2: $4 + 5 \mod 13 = 9$

$\underline{bd}$ h_1: $3 \cdot 4 \cdot 6 \mod 13 = 7$

h_2: $4 + 6 \mod 13 = 10$

$\underline{ca}$, h_1: $3 \cdot 5 \cdot 3 = 6$

h_2: $= 8$

dc = h_1: $3 \cdot 6 \cdot 5 \mod 13 = 12$

h_2: $6 + 5 \mod 13 = 11$

	h1	h2	h(t)	x	g(x)
aa	1	6	0	0	/
ad	2	9	1	1	0
bc	8	9	2	2	3
bd	7	10	3	3	/
ca	6	8	4	4	/
dc	12	11	5	5	/

6	0
7	0
8	4
9	4
10	3
11	0
12	5

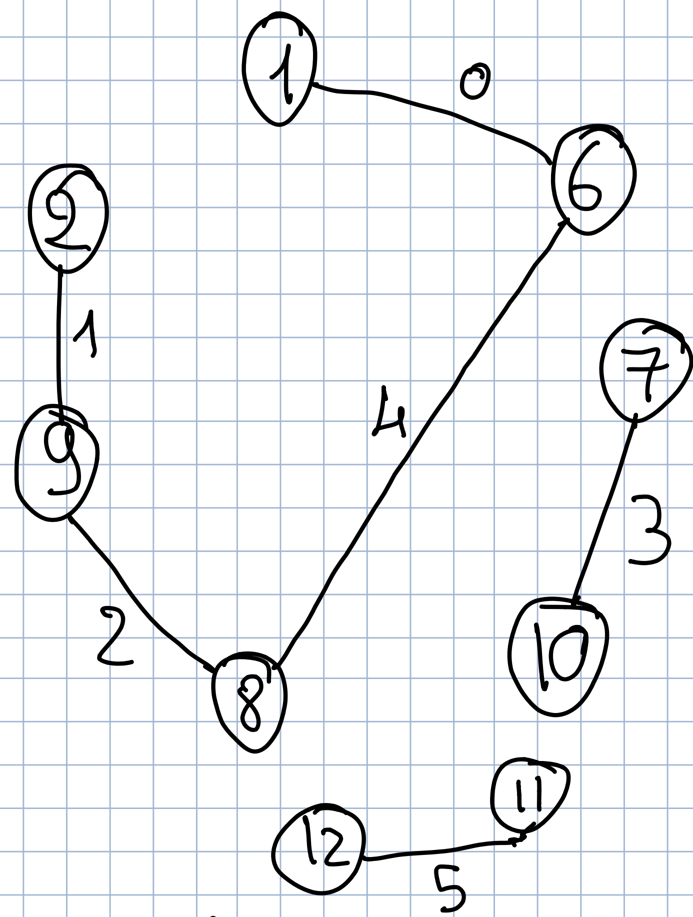
$n = 6$

$h(t) = (g(h_1) + g(h_2)) \bmod n$

$h(t) = (g(h_1) + g(h_2)) \bmod 6$

$g(8) + g(9) \bmod 6$

$4 + 0 \bmod 6$



$g(v_i) = (h(v_{i-1}, v_i) - g(v_i)) \bmod n$

$g(6) = 0 - 0 \bmod 6 = 0$

$g(8) = 4 - 0 \bmod 6 = 4$

$g(9) = 2 - 4 \bmod 6 = -2 \bmod 6 = 4$

$g(2) = 1 - 4 \bmod 6 = -3 \bmod 6 = 3$

$g(10) = 3 - 0 \bmod 6 = 3$

$g(12) = 5 - 0 \bmod 6 = 5$

<u>OK</u>!