```python
taskset = []
n = int(input("enter number of tasks"))
ws = int(input("Enter the number of servers..."))
maxslot = 0
for i in range(n):
    inp = input("enter task_num,profit and deadline :")
    inp = inp.split(" ")
    inp[1] = int(inp[1])
    inp[2] = int(inp[2])
    if(maxslot < inp[2]):
        maxslot = inp[2]
    taskset.append(inp)

taskset.sort(key=lambda taskset: taskset[1], reverse=True)

print("sorting the tasks int the descending order of profits")

for t in taskset:
    print(t)

print("-"*15)
print()

servers = [[0 for i in range(maxslot)] for j in range(ws)]

for i in taskset:
    pos = int(i[2])-1
    if(pos < maxslot and pos >= 0):
        s = -1
        while(1):
            for j in range(len(servers)):
                if servers[j][pos]==0:
                    s = j
                    break
            if s!= -1:
                break
            pos -= 1
        if pos>=0:
            servers[s][pos] = i

for slots in servers:
    for i in range(slots.count(0)):
        slots.remove(0)

opt_profit = 0

for i in servers:
    server_profit = 0
    print("server - ",servers.index(i))
    print(i)
    for t in i:
        server_profit += t[1]
    print("Server Profit: ",server_profit)
    opt_profit += server_profit
    print("-"*15)

print("the task schedule is")
print(servers)
print("-"*15)

print("optimal profit is ", opt_profit)
```

```
deadline :enter task_num,profit and
deadline :enter task_num,profit and
deadline :enter task_num,profit and
deadline :enter task_num,profit and
deadline :enter task_num,profit and
deadline :enter task_num,profit and
deadline :enter task_num,profit and
deadline :enter task_num,profit and
deadline :enter task_num,profit and
deadline :sorting the tasks int the descending
order of profits
['4', 25, 5]
['1', 20, 6]
['7', 20, 2]
['8', 20, 4]
['3', 15, 4]
['9', 15, 7]
['2', 10, 2]
['5', 10, 7]
['10', 10, 3]
['6', 5, 6]
---------------

server -  0
[['7', 20, 2], ['10', 10, 3], ['8', 20, 4], ['4',
25, 5], ['1', 20, 6], ['9', 15, 7]]
Server Profit:  110
---------------
server -  1
[['2', 10, 2], ['3', 15, 4], ['6', 5, 6], ['5',
10, 7]]
Server Profit:  40
---------------
the task schedule is
[[['7', 20, 2], ['10', 10, 3], ['8', 20, 4], ['4',
25, 5], ['1', 20, 6], ['9', 15, 7]], [['2', 10,
2], ['3', 15, 4], ['6', 5, 6], ['5', 10, 7]]]
---------------
optimal profit is  150
```

Enjoying Compiler?

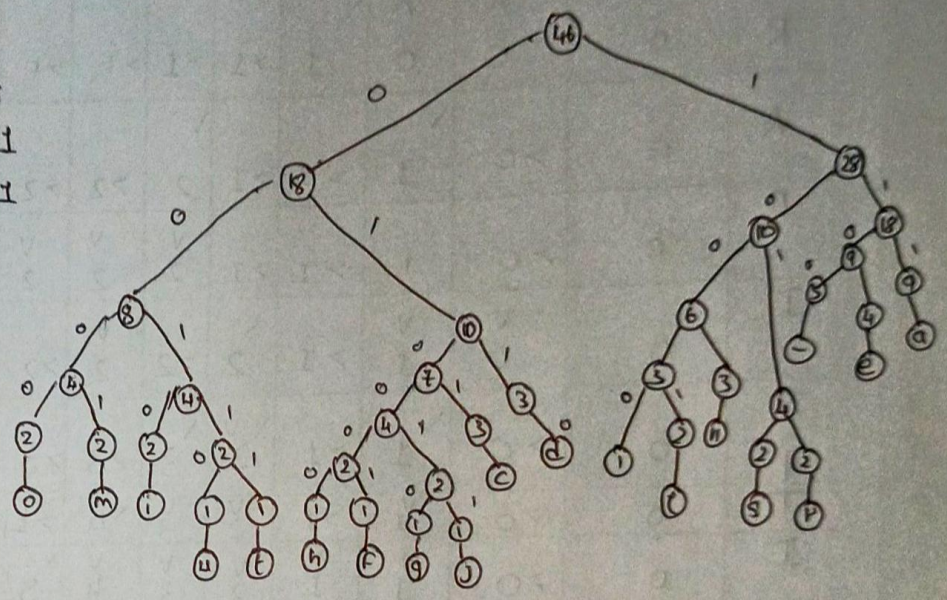
Make your opinion count

RATE THE APP

Kalasalingam Academy of research and education

frequency table

a - 9	g - 1
– - 5	y - 1
n - 3	f - 1
c - 3	h - 1
d - 3	u - 1
e - 4	t - 1
l - 2	
s - 2	
i - 2	
m - 2	
o - 2	
r - 2	
k - 1	



huffman codes:

a = 111	k = 10000
– = 1100	g = 010010
e = 1101	y = 010011
n = 1001	f = 010001
c = 0101	h = 010000
d = 011	u = 00110
l = 10001	t = 00111
s = 1010	
i = 0010	
m = 0001	
o = 0000	
r = 1011	

Compressed string:

10000 1111 0001 1111 1010 111 10001 0010 1001 0100 10
1 0010 1111 0001 111 0011 010 1111 0111 1 0100 1010011
1100 00 00010001 1100 10 11110 1101 0110 1111 0110101
01000 1100 1111 0010 1111 1100 11 0101 1001 1001 0111
00 111 0010 0000 1001

String size = 184

2. a) RADIATION & VARIATION

	E	V	A	R	I	A	T	I	O	N
E	0	0	0	0	0	0	0	0	0	0
R	0	^ 0	^ 0	\ 1	>1	>1	>1	>1	>1	>1
A	0	>0	\ 1	>1	>1	\ 2	>2	>2	>2	>2
D	0	>0	1	>1	>1	v 2	v 2	v 2	v 2	v 2
I	0	v 0	v 1	>1	\ 2	>2	v 2	>2	v 2	v 2
A	0	>0	\ 1	>1	\ v 2	\ 3	>3	>3	>3	>3
T	0	>0	v 1	>1	v 2	v 3	\ 4	>4	>4	>4
I	0	>0	v 1	v 1	v 2	v 3	v 4	\ 5	>5	>5
O	0	>0	v 1	v 1	v 2	v 3	v 4	v 5	\ 6	>6
N	0	>0	v 1	v 1	v 2	v 3	v 4	v 5	v 6	\ 7

b) ALGORITHMS & LOGARITHMS

	ε	L	O	G	A	R	I	T	H	M	S
ε	0	0	0	0	0	0	0	0	0	0	0
A	0	> 0	0	0	\ 1	> 1	> 1	> 1	> 1	> 1	> 1
L	0	\ 1	> 1	> 1	> 1	> 1	> 1	> 1	> 1	> 1	> 1
G	0	1	1	\ 2	> 2	> 2	> 2	> 2	> 2	> 2	> 2
O	0	1	\ 2	2	2	2	2	2	2	2	2
R	0	1	2	2	2	\ 3	> 3	> 3	> 3	> 3	> 3
I	0	1	2	2	2	3	\ 4	> 4	> 4	> 4	> 4
T	0	1	2	2	2	3	4	\ 5	> 5	> 5	> 5
H	0	1	2	2	2	3	4	5	\ 6	> 6	> 6
M	0	1	2	2	2	3	4	5	6	\ 7	> 7
S	0	1	2	2	2	3	4	5	6	7	\ 8

3.

	1	2	3	4
Key	Rank	Name	Email	Regno
Freq	5	3	6	7

	0	1	2	3	4
0	0	5	11^1	25^1	39^3
1		0	3	11	26^3
2			0	6	19^4
3				0	7
4					0

$c(0,1) = 5$

$c(1,2) = 3$

$c(2,3) = 6$

$c(3,4) = 7$

$$c(0,2) = \min \begin{cases} c(0,0) + c(1,2) + w(0,2) \\ c(0,1) + c(2,2) + w(0,2) \end{cases}$$

$$= \min \begin{cases} 0+3+8 \\ 5+0+8 \end{cases}$$

$$= 11^1$$

$$c(1,3) = \min \begin{cases} c(1,1) + c(2,3) + w(1,3) \\ c(1,2) + c(3,3) + w(1,2) \end{cases}$$

$$= \min \begin{cases} 0+6+9 \\ 3+0+9 \end{cases}$$

$$= 11^3$$

$$c(2,4) = \begin{cases} c(2,2) + c(3,4) + w(2,4) \\ c(2,3) + c(4,4) + w(2,4) \end{cases}$$

$$= \begin{cases} 0+7+13 \\ 6+0+13 \end{cases}$$

$$= 19^4$$

$$c(0,3) = \min \begin{cases} c(0,0) + c(1,3) + w(0,3) \\ c(0,1) + c(2,3) + w(0,3) \\ c(0,2) + c(3,3) + w(0,3) \end{cases}$$

$$= \min \begin{cases} 0+11+14 \\ 5+6+14 \\ 11+0+14 \end{cases}$$

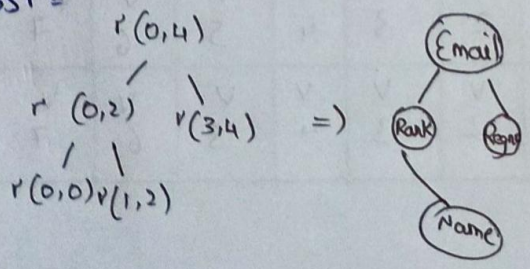
$$= 25$$

$$c(1,4) = \min \begin{cases} c(1,1) + c(2,4) + w(1,4) \\ c(1,2) + c(3,4) + w(1,4) \\ c(1,3) + c(4,4) + w(1,4) \end{cases}$$

$$= \min \begin{cases} 0+19+16 \\ 3+7+16 \\ 11+0+16 \end{cases} = 26^3$$

$$c(0,4) = \min \begin{cases} c(0,0) + c(1,4) + w(0,4) \\ c(0,1) + c(2,4) + w(0,4) \\ c(0,2) + c(3,4) + w(0,4) \\ c(0,3) + c(4,4) + w(0,4) \end{cases} = \min \begin{cases} 0+26+21 \\ 5+19+21 \\ 11+7+21 \\ 25+0+21 \end{cases} = 39^3$$

OBST =



4)

keys	10	20	30	40
freq	2	6	3	4

	0	1	2	3	4
0	0	2	10^2	16^2	27^2
1		0	6	12^2	23^2
2			0	3	10^4
3				0	4
4					0

$C(0,1) = 2 \quad C(1,2) = 6$

$C(2,3) = 3 \quad C(3,4) = 4$

$$C(0,2) = \min \begin{cases} C(0,0) + C(1,2) + w(0,2) \\ C(0,1) + C(2,2) + w(0,2) \end{cases}$$

$$= \min \begin{cases} 0 + 6 + 8 \\ 2 + 0 + 8 \end{cases}$$

$$= 10^2$$

$$C(1,3) = \min \begin{cases} C(1,1) + C(2,3) + w(1,3) \\ C(1,2) + C(3,3) + w(1,3) \end{cases}$$

$$= \min \begin{cases} 0 + 3 + 9 \\ 6 + 0 + 9 \end{cases}$$

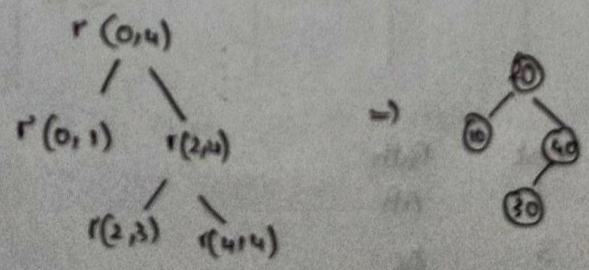
$$= 12^2$$

$$C(0,3) = \min \begin{cases} C(0,0) + C(1,3) + w(0,3) \\ C(0,1) + C(2,3) + w(0,3) \\ C(0,2) + C(3,3) + w(0,3) \end{cases} = \min \begin{cases} 0 + 12 + 11 \\ 2 + 3 + 11 \\ 10 + 0 + 11 \end{cases} = 16^2$$

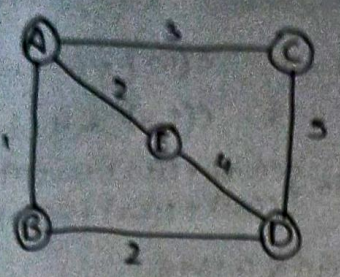
$$C(1,4) = \min \begin{cases} C(1,1) + C(2,4) + w(1,4) \\ C(1,2) + C(3,4) + w(1,4) \\ C(1,3) + C(4,4) + w(1,4) \end{cases} = \min \begin{cases} 0 + 10 + 13 \\ 6 + 4 + 13 \\ 12 + 0 + 13 \end{cases} = 23^3$$

$$C(0,4) = \min \begin{cases} C(0,0) + C(1,4) + w(0,4) \\ C(0,1) + C(2,4) + w(0,4) \\ C(0,2) + C(3,4) + w(0,4) \\ C(0,3) + C(4,4) + w(0,4) \end{cases} = \min \begin{cases} 0 + 23 + 15 \\ 2 + 10 + 15 \\ 10 + 4 + 15 \\ 16 + 0 + 15 \end{cases} = 27^2$$

OBST



5)



from A

	A	B	C	D	E	Choose vertex
int ()	0	∞	∞	∞	∞	A
{A}	–	1,A	3,A	∞	2,A	B
{A,B}	–	–	3,A	2,D 3,A	2,A	E
{A,E}	–	1,A	3,A	4,E 6,A 3,A	–	C
{A,C}	–	1,A	–	3,A	2,A	D
{A,B,D}	–	–	3,A	–	2,A	

Result :

	Cost	Path
A-B :-	1	AB
A-C :	3	AC
A-D :	3	ABD
A-E :	2	AE

6.

Tasks	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9
Profit	20	25	10	15	9	22	19	22	30
Deadline	3	3	4	5	7	7	6	2	2

Sorted tasks with respect to profit

Task	Profit	Deadline
T_9	30	2
T_2	25	3
T_6	22	7
T_8	22	2
T_1	20	3
T_7	19	6
T_4	15	5
T_3	10	4
T_5	9	7

Max deadline = 7

Slots:

1	2	3	4	5	6	7
T_8	T_9	T_2	T_3	T_4	T_7	T_6

Task schedule = $T_8\ T_9\ T_2\ T_3\ T_4\ T_7\ T_6$

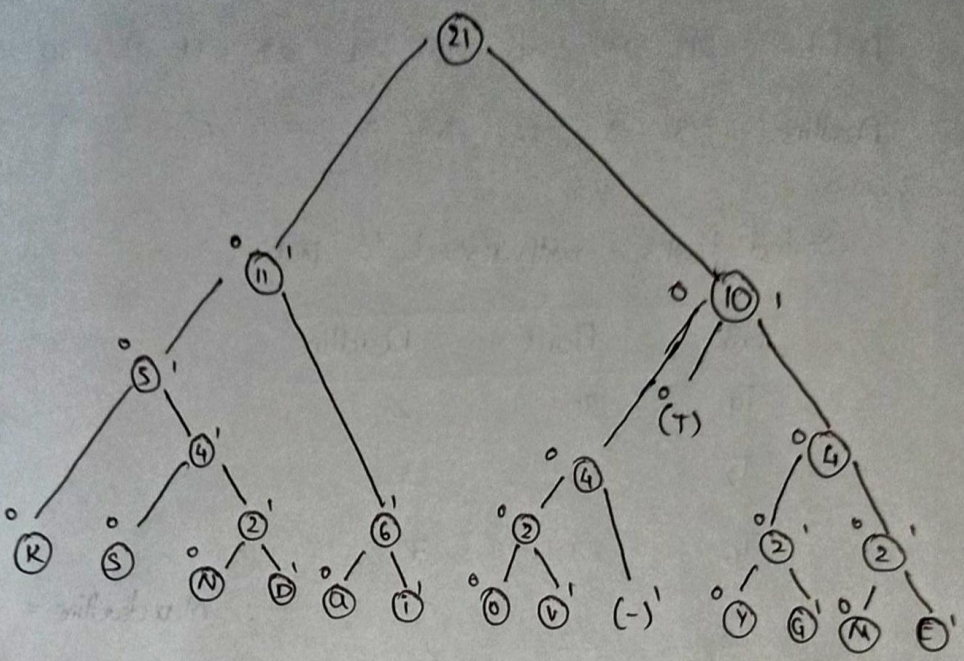
profit = $22 + 30 + 25 + 10 + 15 + 19 + 22$

$= 143$

KAMISETTY GOVINDA SAI

frequency table:

a - 3
i - 3
'-' - 2
S - 2
T - 2
K - 1
M - 1
E - 1
Y - 1
G - 1
O - 1
V - 1
N - 1
D - 1



huffman codes

D - 00111
N - 00110
V - 1001
O - 1000
G - 1101
Y - 1100
E - 1111
M - 1110
K - 000
T - 10
S - 0010
'-' - 101
i - 011
a - 010

Compressed string:

000 010 1110 011 0010 1111 1010110010 1
110 1000 1001 011 00 110 00111010
101 00 10 010 011

Compressed string Size = 73