Maharashtra Academy of Engineering & Educational Research's
MAHARASHTRA INSTITUTE OF TECHNOLOGY, PUNE

ASSIGNMENT / TEST BOOKLET

Student's Name : ....

Class : .... Division : .... Roll No.: .... Academic Year : 20 -20

Subject : .... Assignment / Test No. : .... Date : ....

PLEDGE

I solemnly affirm that I have written this Assignment/Test based on my own preparation. I have neither copied it from others nor given it to others for coping. I know that this is to be submitted as a part of my submission at the end of the term.

Signature of the student

Q. No.	1	2	3	4	5	6	7	8	9	10	Total
Marks/Grade											

Name & sign of the faculty Member

(Please start writing assignment/ test from here)

0/1 Knapsack Problem- BB.

Problem statement

The 0/1 knapsack problem states that -
There are 'n' objects given & capacity of knapsack is 'm'.
Then select some objects to fill the knapsack in such a way that it should not exceed the capacity of knapsack & maximum profit can be earned.
The knapsack problem is a miximization problem.
That means we will always seek for maximum P_iX_i (where P_i represents profit of object X_i)

We can get ΣP_iX_i maximum iff $-\Sigma P_iX_i$ is minimum minimize profit $- \sum_{i=1}^{n} p_i x_i$

subject to $\sum_{i=1}^{n} w_i x_i$

such that $\sum w_i x_i <= m$ & $x_i = 0$ or 1
where $1 \leq i \leq n$

→ we will discuss the BB strategy for 0/1 knapsack problem -using fixed tuple size formulation.

→ we will design the state space tree & compute $c^\wedge(\cdot)$ & $u(\cdot)$
where $c^\wedge(x)$ – represent the ~~approx~~ approximate cost used for computing the least cost $c(x)$.

– clearly, $u(x)$ denotes the upper bound. As we know, upper bound is used to kill those nodes in the state space tree, which can not lead to the answer node.

let x, be the node at level j. then, we will draw the state space tree for fixed tuple formulation having levels $1 \leq j \leq n+1$.

Then, we need to compute $\hat{c}(x)$ & $u(x)$ such that $\hat{c}(x) \leq c(x) \leq u(x)$ for each node.

LC BB sol^n

→ The LC BB sol^n can be obtained using fixed tuple size formulation.
→ The steps to be followed for LCBB sol^n are
1. Draw state space tree
2. compute $\hat{c}(\cdot)$ & $u(\cdot)$ for each node
3. If $\hat{c}(x) >$ upper, kill node x
4. otherwise the minimum cost $\hat{c}(x)$ becomes E-node.
Generate, children for E-node.
5. Repeat 3 & 4 steps untill all the nodes get covered.
6. The minimum cost $\hat{c}(x)$ becomes the answer node.

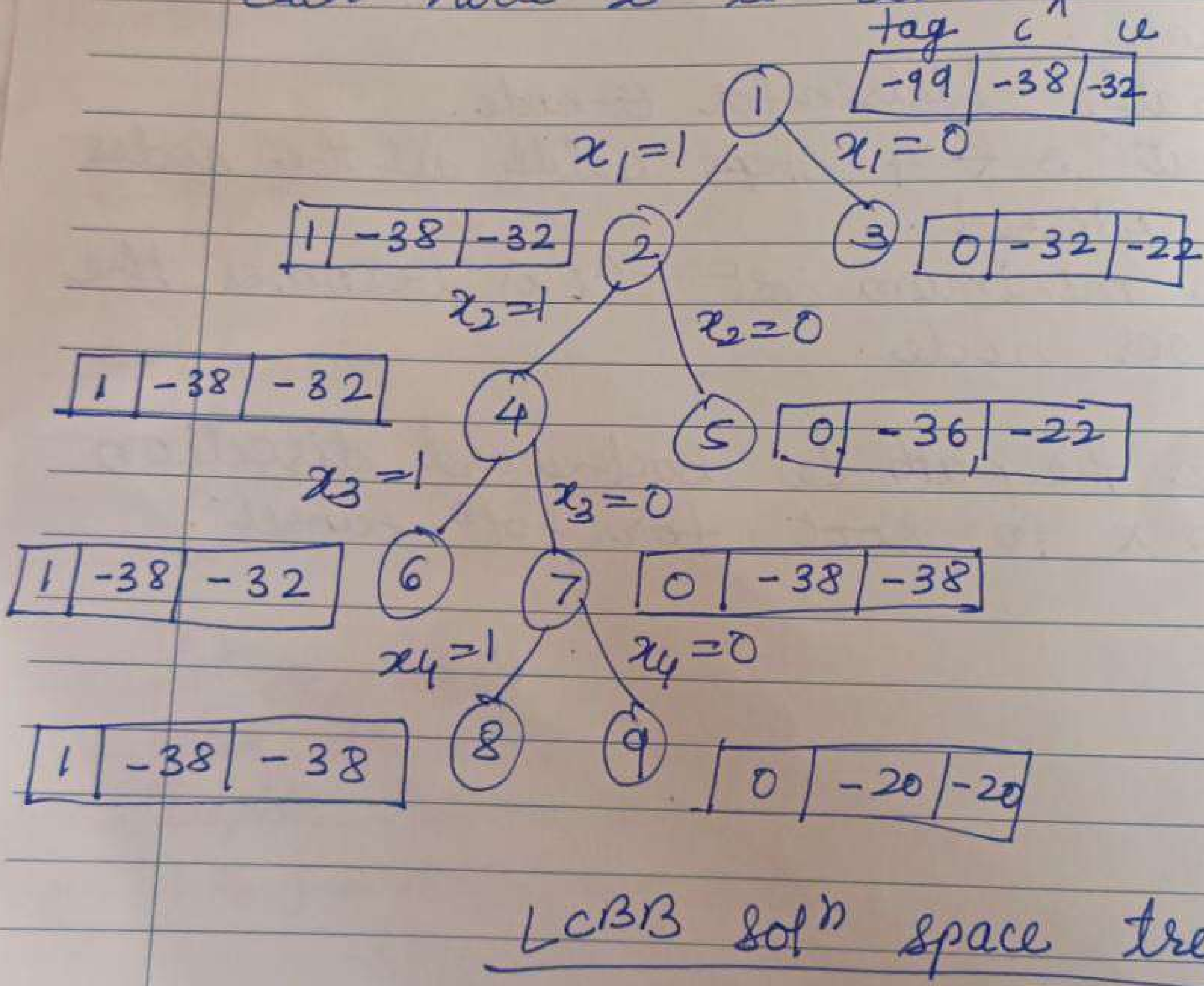
Trace the path in backward direction from x to root for sol^n subset.

ex consider Knapsack instance $n=4$ with capacity $m = 15$. such that

object i	p_i	w_i
1	10	2
2	10	4
3	12	6
4	18	~~8~~ 9

sol^n let us design state space tree using fixed tuple size formulation.

The computation of $\hat{c}(x)$ & $u(x)$ for each node x is done.



LCBB sol^n space tree

Assignment/Test Page No

At each node a structure is drawn in which computation of $\hat{c}(\cdot)$ & $u(\cdot)$ is given.

The tag field is useful for tracing the path.

$$u(1) = -\sum p_i \,, \quad i = 1, 2, 3.$$
$$= -(10+10+12)$$
$$= -32.$$

(if we select $i=4$, it exceed the capacity).

$$\therefore \hat{c}(1) = u(1) - \left[\frac{m - \text{current total wt}}{\text{Actual wt of remaining obj}}\right] * [\text{Actual pf. of remaining object}]$$

$$= -32 - \left[\frac{15-(2+4+6)}{9}\right] * 18$$

$$= -32 - \left[\frac{3}{9} * 18\right]$$

$$= -32 - [6]$$

$$\hat{c}(1) = -38$$

→ In this way, considering each possibility of object being in knapsack or not being in knapsack.

$c\hat{}(x)$ & $u(x)$ is computed.
Each time minimum $-\sum p_i x_i$ will become E-node and we will get the answer node as node 8.

If we trace the tag field we will get tag(2) – tag(4) – tag(7) – tag(8) ie. 1101

Hence, $x_4 = 1$, $x_3 = 0$, $x_2 = 1$ & $x_1 = 1$.

We will select object x_4, x_2 & x_1 to fill up the knapsack & gain profit.

$$u(2) = -\sum P_i \quad , \; i = 1, 2, 3$$
$$= -(10 + 10 + 12)$$
$$u(2) = -32$$

$$c\hat{}(2) = u(2) - \left[\frac{15 - (2+4+6)}{9}\right] * 18$$
$$= -32 - \frac{3}{9} * 18$$
$$= -38.$$

$$u(3) = -\sum P_i \quad i = 2, 3$$
$$= -(10 + 12) = -22$$
$$c\hat{}(3) = u(3) - \left[\frac{15 - (4+6)}{9}\right] * (\cancel{10+18})\, 18$$
$$= -22 - \left[\frac{15 - 10}{9} * 18\right]$$

Assignment/Test Page No.

$= -22 - \left(\frac{5}{9} * 18\right)$

$= -22 - 10 = -32$

$u(4) = -\Sigma P_i \quad i = 1, 2, 3.$

$= -32$

$c^\wedge(4) = -38$

$u(5) = -\Sigma P_i \quad i = 1, 3.$

$= -\cancel{\Sigma}(10+12)$

$= \underline{-22}$

$c^\wedge(5) = u(5) - \left[\frac{15-(2+6)}{9}\right] * 18$

$= -22 - \left(\frac{15-8}{9} * 18\right)$

$= -22 - (7 * 2)$

$= -22 - 14$

$= \underline{-36.}$

$u(6) = -\Sigma P_i, \quad i = 1, 2, 3.$

$= -32$

$c^\wedge(6) = -38$

Assignment/Test Page No.

$$u(7) = -\Sigma P_i \quad , i=1,2,4$$
$$= -(P_1+P_2+P_4)$$
$$= -(10+10+18)$$
$$= -38$$

$$c\hat{}(7) = u(7) - \left[\frac{15-(2+4+9)}{6} * 12\right]$$
$$= -38 - \left(\frac{15-15}{6} * 12\right)$$
$$= -38 - 0$$
$$= -38$$

$$u(8) = -\Sigma P_i \quad , i=1,2,4$$
$$= -38$$

$$c\hat{}(8) = -38$$

$$u(9) = -\Sigma P_i \quad i=1,2$$
$$= -(P_1+P_2)$$
$$= -(10+10) = -20$$

$$c\hat{}(9) = u(9) - \left[\frac{15-(2+4+9)}{6} * 12\right]$$
$$= -20 - (0)$$
$$= -20.$$

Maharashtra Academy of Engineering & Educational Research's
MAHARASHTRA INSTITUTE OF TECHNOLOGY, PUNE
ASSIGNMENT / TEST BOOKLET

Student's Name : ....

Class : .... Division : .... Subject : .... Roll No. : .... Academic Year : 20 -20

Assignment / Test No. : .... Date : ....

PLEDGE

I solemnly affirm that I have written this Assignment/Test based on my own preparation. I have neither copied it from others nor given it to others for coping. I know that this is to be submitted as a part of my submission at the end of the term.

Q. No.	1	2	3	4	5	6	7	8	9	10	Total
Marks/Grade											

Signature of the student

Name & sign of the faculty Member

(Please start writing assignment/ test from here)

FIFO Branch & Bound sol^n

→ The space tree with variable tuple size formulation can be drawn &
→ c^(·) & u(·) is computed.

→ Initially, upper = u(1) = −32.
Then children of node 1 are generated.

→ Node 2 becomes E-node & hence children 4 & 5 are generated.

Node 4 & 5 are added in the list of live nodes.

Next, node 3 becomes E-node & children 6 & 7 are generated.

Assignment/Test Page No.

As $c^\wedge(7) >$ upper ans. we will kill node 7.
Hence node 6 will be added in the list of lives nodes.

Node 4 is E-node & children 8 & 9 are generated.
The upper is updated & it is now $u(9) = -38$.
nodes 8 & 9 are added in the list of live nodes.

Node 5 & 6 becomes the next E-node but as $c^\wedge(5) >$ upper &
$c^\wedge(6) >$ upper
$\therefore$ kill nodes 5 & 6.

Now, 8 becomes next E node. & children ~~8~~10 & 11 are generated.

As node 10 is infeasible, $\therefore$ do not consider it.

$c^\wedge(11) >$ upper, Hence kill node 11.

Node 9 becomes next E-node. & upper $= -38$.

children 12 & 13 are generated.
But $\hat{c}(13) >$ upper, so kill node 13.

$\therefore$ Finally, node 12 becomes an answer node

$\therefore$ sol^n $= \{x_1 = 1, x_2 = 1, x_3 = 0, x_4 = 1\}$

ex 2 :- Draw the portion of state space tree generated by LC-KNAP for the knapsack instance —

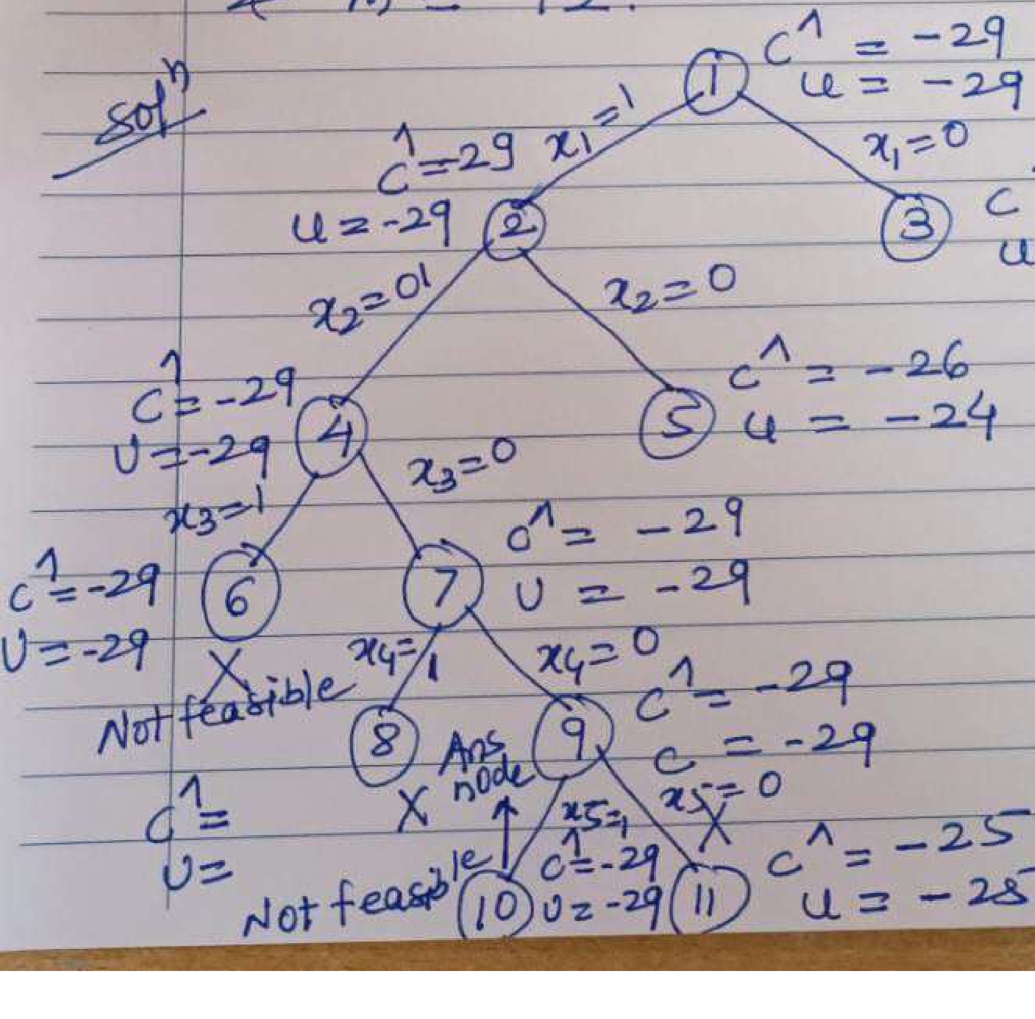
$n = 5$

$(P1, P_2, \cdots P_5) = (10, 15, 6, 8, 4)$

$(W1, W_2, \cdots W_5) = (4, 6, 3, 4, 2)$

& $m = 12$.

sol^n



sol^n $= \{x_1, x_2, x_3, x_4, x_5\}$
$= \{1, 1, 0, 0, 1\}$

FIFO Branch & Bound (*) example

space tree for variable tuple size formulation –

ex ① $n = 4$, capacity $m = 15$

object i	P_i	W_i
1	10	2
2	10	4
3	12	6
4	18	9

sol^n