Lab-9

Aim: Knapsack problem by using Genetic Algorithm

Item	weight(kg)	value
A	5	12
B	3	5
C	7	10
D	2	7

→ There are four items. Each item is associated with some weight(w) and value at item(v).

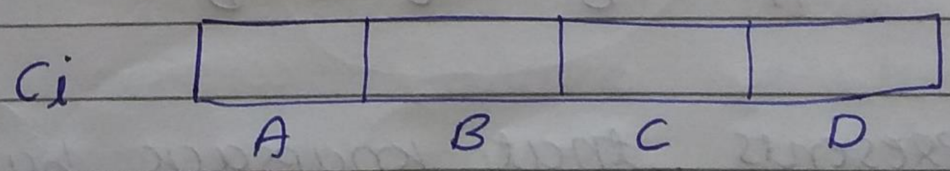
→ There is a knapsack K with limited capacity that can hold almost 12 kg

problem

The problem is that which item should be kept in the knapsack so as it will maximizes knapsack value without breaking knapsack.

Step1 Chromosomes Encoding

consider an array C_i represents all items together



Gene 0 - represents absence of item in the knapsack

1 - represents presence of item in the knapsuck

4-bits are requested to represent chromosomes Encoding

set space = 2^4

Initial population is created and chromosomes randomly created.

Generation 1

C_1	0	1	1	0
C_2	0	1	0	1
C_3	1	1	0	1
C_4	1	1	1	1

Step2 fitness function

- Next step is to determine fitness fuc^n which is used to evaluate how good particular solution is

Lets take C_1

0	1	1	0
A	B	C	D

represents that knapsack has presence of item B & C and absense of item A & D

value of knapsack = value of B + value of C
= 5 + 10
= 15

weight of knapsack = weight of B + weight of item C
= 3 + 7
= 10 kg

knapsack capacity = 12 kg
as 12kg > 10kg

so, C_1 is accepted

→ similary check for C_2, C_3, C_4

	Generation 1				fitness	
					W	V
C_1	0	1	1	0	10	15
C_2	0	1	0	1	5	12
C_3	1	1	0	1	10	24
C_4	1	1	1	1	17	34

step3: selection

→ Next step is to collect the fittes individual and wake up the next generation chromosome selection

By using Roulette wheel

- spin the Roulette wheel and whenever the wheel stops, the individual gets selected at that point.

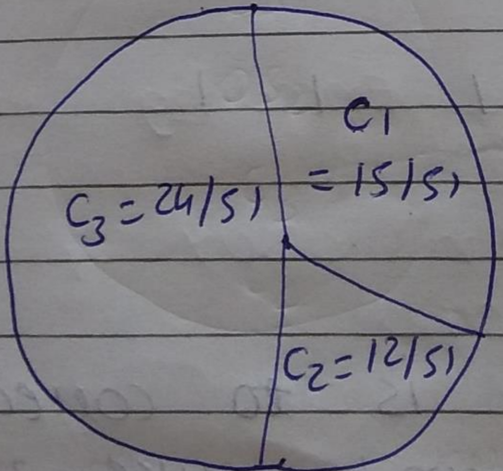
- The individual that has the highest fitness value gets larger share of the wheel

e.g. Total fitness value = $15+12+24+0$
$= 51$

fitness value of $C_3 = 24$; largest fitness

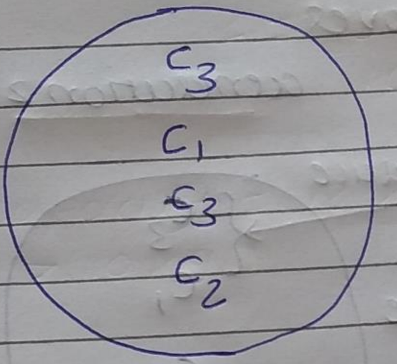
so, C_3 occupies half of the wheel as $24/51$

C_4 has zero chance of wining, C_3 has the highest probability of getting selected in the next generation



→ After spining Roulette wheel, in the first spin C_3 will be selected and then C_1 after that C_3 and C_2

→ Individuals of next generation are selected as follows:



Generation 2

Step4 crossover

→ The crossover operation takes the selected chromosomes for mating and mixes the genetic material to produce offspring

C_3	1	1	0	1
C_1	0	1	1	0

one-point crossover: Randomly select the position on the chromosomes about which gene would be exchange

C_3	1	1	0	1
C_1	0	1	1	0

← crossover point

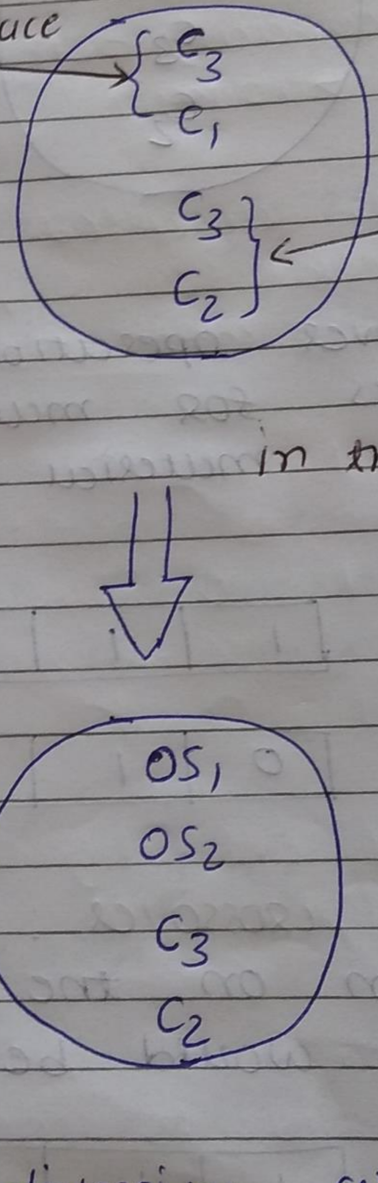
Result of one point crossover
i.e. produced offspring

OS,	1	1	0	0
OS_2	0	1	1	1

- After crossover

Generation 2

OS, will replace C_3 and OS_2 will replace C_1



C_3 and C_2 do not go for the crossover so will remain in the next generation

~~Generation 3~~

step 5 Mutation

- Providing diversity within population so that search algorithm does not nessasarily get stuck at local maxima

Before applying mutation

OS_1	$\underline{1}$	1	0	0
OS_2	0	$\underline{1}$	1	1
C_3	1	1	0	1
C_2	0	1	0	1

After applying mutation

					fitness	
					w	v
OS_1	0	1	0	0	3	5
OS_2	0	0	1	1	9	17
C_3	1	1	0	1	10	24
C_2	0	1	0	1	5	12

$\therefore$ Total fitness value $= 5+17+24+12$

$= 58$