Nome-Nikita Agarwal ROU NO-CSML9030 accept a, aur [100], element initialize bottom, (arr [middle] bottern = middle -1 bottom = 0 top = m - 1 middle = (bottom +top)/2 top= retwen yes 3 middle +1 middle = dements middle / bottom +top)/acturn -1 II Cyclomatic complexity: 1st way: - V(01)= e-m+2P ulners, e -, edges P -> connected components (For a single program, P=1) LLW, e= 14, m=12, P=1 ··· V(01) = 14-12+2(1) A + 1, where a is the producate node 2nd way, U(01) = have ture conditions. tlere, me 80, T=3 - " V(a) = 3+1=4.

2 d way, U(G1) = no. of jugians.

Region 1 = 7,8,10,9,7

Region 2 = 5,7,8,10,6,5

Region 3 = 12,4,5,6,10,12

Region 4 = the entire & CON CECH

Region 4 = the entire & CON CECH

1 V(V1) = 4.

Cyclomatic Complexity is a structural testing method that uses the source code of a program to find every possible executable path. In Path Testing, it is used to delumine the number of linearly independent paths and then test cases are generated for each path.

1 2000	INPUTS		Expected	Remarks
PATHS	arr[100]	element	output	
Path 1:	Cempty	2	Failure	tempty list.
1-2-3-12-4-11 Path1:	[1,2,3,4,5]	6	Failure	Clement not in arrived
1-2-3-12-4-11 Parth 2: 1-2-3-12-4-5-6-10	[1,2,3,415]	4	Repeat and Success	arm [100]
Path 3:	[1,2,3,4,5]	2	Repeat and Success	Glement in aru [100]
Podu 4: 1-2-3-12-4-5-9-10	[1,2,3,4,5]	3	Sueces,	Celement in aur [100]