**Case Study**:Determine the cyclomatic complexity of this program. for the following **Binary Search program**

|  |  |
| --- | --- |
| 01 | #include <stdio.h> |

|  |  |
| --- | --- |
| 02 |  |

|  |  |
| --- | --- |
| 03 | int |

|  |  |
| --- | --- |
| 04 | main(int argc, char \*\*argv) |

|  |  |
| --- | --- |
| 05 | { |

|  |  |
| --- | --- |
| 06 | int a[] = {2, 4, 6, 8, 10, 12, 14, 16}; |

|  |  |
| --- | --- |
| 07 | int key = atoi(argv[1]); |

|  |  |
| --- | --- |
| 08 | int n = 8; |

|  |  |
| --- | --- |
| 09 | int low = 0; |

|  |  |
| --- | --- |
| 10 | int high = n - 1; |

|  |  |
| --- | --- |
| 11 | int middle; |

|  |  |
| --- | --- |
| 12 | int flag = 0; |

|  |  |
| --- | --- |
| 13 |  |

|  |  |
| --- | --- |
| 14 | while (low <= high) { |

|  |  |
| --- | --- |
| 15 | middle = (low + high) / 2; |

|  |  |
| --- | --- |
| 16 | if (key == a[middle]) { |

|  |  |
| --- | --- |
| 17 | printf("Found key %d at position %d\n", key, middle); |

|  |  |
| --- | --- |
| 18 | flag = 1; |

|  |  |
| --- | --- |
| 19 | } |

|  |  |
| --- | --- |
| 20 | else if (key < a[middle]) { |

|  |  |
| --- | --- |
| 21 | high = middle - 1; |

|  |  |
| --- | --- |
| 22 | } |

|  |  |
| --- | --- |
| 23 | else { |

|  |  |
| --- | --- |
| 24 | low = middle + 1; |

|  |  |
| --- | --- |
| 25 | } |

|  |  |
| --- | --- |
| 26 | if (flag) |

|  |  |
| --- | --- |
| 27 | break; |

|  |  |
| --- | --- |
| 28 | } |

|  |  |
| --- | --- |
| 29 |  |

|  |  |
| --- | --- |
| 30 | if (! flag) |

|  |  |
| --- | --- |
| 31 | printf("Key %d was not found!\n", key); |

|  |  |
| --- | --- |
| 32 |  |

|  |  |
| --- | --- |
| 33 | return 0; |

|  |  |
| --- | --- |
| 34 } |  |
|  |  |