**Linear Search**

**LINEAR\_SEARCH(A, N, VAL)**

1. Set POS=-1
2. SET I=1
3. Repeat step 4 while I<=N
4. IF A[I]=VAL

SET POS=I

PRINT POS

Goto Step 6

[End of if]

Set I=I+1

[End of loop]

1. IF POS=-1

PRINT “Value does not exist”

[End of if]

1. Exit

**Binary Search**

**BINARY\_SEARCH(A, lower\_bound, upper\_bound, VAL)**

1. Set BEG=lower\_bound, END=upper\_bound,POS=-1
2. Repeat steps 3 and 4 while BEG<=END
3. Set MID=(BEG+END)/2
4. IF A[MID]=VAL

Set POS=MID

PRINT POS

Goto Step 6

Else if A[MID]>VAL

Set END=MID-1

Else

Set BEG=MID+1

[End of if]

[End of loop]

1. If POS=-1

PRINT “Value does not exist”

[End of if]

1. Exit

**Interpolation Search**

**INTERPOLATION\_SEARCH (A, lower\_bound, upper\_bound, VAL)**

1. [INITIALIZE] SET LOW = lower\_bound,HIGH = upper\_bound, POS = –1
2. Repeat Steps 3 to 4 while LOW <= HIGH
3. SET MID = LOW + (HIGH – LOW) ×((VAL – A[LOW]) / (A[HIGH] – A[LOW]))
4. IF VAL = A[MID] POS = MIDPRINT POS Go to Step 6

ELSE IF VAL < A[MID]SET HIGH = MID – 1

ELSE

SET LOW = MID + 1

[END OF IF]

[END OF LOOP]

1. IF POS = –1 PRINT " Value does not exist "

[END OF IF]

1. EXIT

**Jump Search**

**JUMP\_SEARCH (A, lower\_bound, upper\_bound, VAL, N)**

1. Set STEP=sqrt(N),I=0,LOW=lower\_bound, HIGH=upper\_bound,POS=-1
2. Repeat Step 3 while I < STEP
3. IF VAL < A[STEP]SET HIGH = STEP – 1 ELSESET LOW = STEP + 1 [END OF IF] SET I = I + 1 [END OF LOOP]
4. SET I = LOW
5. Repeat Step 6 while I <= HIGH
6. IF A[I] = Val

POS = I

  PRINT POS Go to Step 8

[END OF IF]

SET I = I + 1

[END OF LOOP]

1. IF POS = –1 PRINT " Print" Value does not exist ""

[END OF IF]

1. EXIT