**EXPERIMENT:**

Create an array by reading the inputs from file ‘numbers.txt’ and find kth smallest element in an array.

**PSEUDO CODE**

1. Create a swap function.
2. Create heapify function

Heapify(A, i) {

le <- left(i)

ri <- right(i)

if (le<=heapsize) and (A[le]>A[i])

largest <- le

else

largest <- i

if (ri<=heapsize) and (A[ri]>A[largest])

largest <- ri

if (largest != i)

{

exchange A[i] <-> A[largest]

Heapify(A, largest)

}

}

1. Create BuildHeap fuction

BuildHeap(A)

{

heapsize <- length(A)

for i <- floor( length/2 ) downto 1

Heapify(A, i)

}

1. Create Heap Sort function

Heapsort(A) {

BuildHeap(A)

for i <- length(A) downto 2 {

exchange A[1] <-> A[i]

heapsize <- heapsize -1

Heapify(A, 1)

}

1. In Main function

Open file in read mode

Read each number line by line and Insert it in an array

Call the Heap Sort fuction

Print the Sorted Array

Insert the kth smallest term we want to find

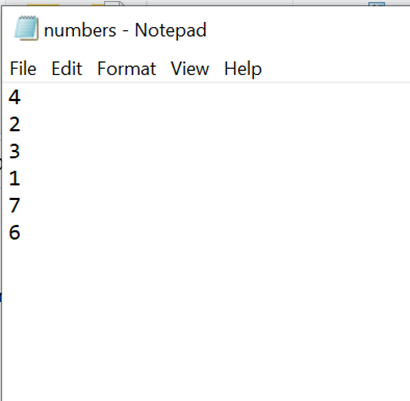
If the kth term is more that of total terms in the array or negative or zero

Print Not valid Kth term

Else

Print the value of arr [ k-1 ]

**INPUT**



**OUTPUT**

