Q1 h) Implement Heap Sort Algorithm. Analyze its time complexity.

TIME COMPLEXITY:

1. Best Case: O(n * log(n))
2. Worst Case: O(n * log(n))
3. Average Case: O(n * log(n))

SPACE COMPLEXITY: O(1)

```
#include <iostream>
#include <vector>
#include <random>


using namespace std;


void inplaceHeapSort(vector<int> & input)
{
    int n = int(input.size());
    for (int i = 0; i < n; i++)
    {
        int childIndex = i;
        while(childIndex > 0)
        {
            int parentIndex = (childIndex - 1) / 2;
            if(input[childIndex] < input[parentIndex])
                swap(input[childIndex], input[parentIndex]);
            else
                break;
            childIndex = parentIndex;
        }
    }
    int size = n;
    while (size > 1)
    {
        swap(input[0], input[--size]);
        int Pindex = 0;
        int LCI = 2 * Pindex + 1;
        int RCI = 2 * Pindex + 2;
        int minIndex = Pindex;
        while (LCI < size)
        {
            if(input[minIndex] > input[LCI])
                minIndex = LCI;
            if(RCI < size && input[minIndex] > input[RCI])
                minIndex = RCI;
            if(minIndex == Pindex)
                break;
            swap(input[Pindex], input[minIndex]);
            Pindex = minIndex;
            LCI = 2 * Pindex + 1;
            RCI = 2 * Pindex + 2;
```

```
        }
    }
}

int main()
{
    vector<int> A(100);
    for(int i = 0; i < 100; ++i)
        A[i] = rand() % 750;
    cout << "INITIAL STATE OF ARRAY:\n";
    for(int i = 0; i < 100; ++i)
        cout << A[i] << " ";
    cout << endl << endl;
    inplaceHeapSort(A);
    reverse(A.begin(), A.end());
    cout << "SORTED ARRAY:\n";
    for(int i = 0; i < 100; ++i)
        cout << A[i] << " ";
    cout << endl;
}
```

OUTPUT:

```
INITIAL STATE OF ARRAY:
307 499 323 158 430 272 294 128 173 709 690 665 492 542 237 503 577 229 340
 112 303 169 459 407 310 183 99 528 566 85 97 326 512 517 560 633 229 649
 579 321 217 422 643 336 485 245 728 91 444 607 251 653 208 694 418 740 624
 446 30 403 472 166 549 524 551 103 477 408 228 433 298 481 385 13 615 64
 563 36 425 419 615 94 129 1 17 695 105 404 201 48 438 623 5 382 252 566
 466 87 188 144

SORTED ARRAY:
1 5 13 17 30 36 48 64 85 87 91 94 97 99 103 105 112 128 129 144 158 166 169
 173 183 188 201 208 217 228 229 229 237 245 251 252 272 294 298 303 307
 310 321 323 326 336 340 382 385 403 404 407 408 418 419 422 425 430 433
 438 444 446 459 466 472 477 481 485 492 499 503 512 517 524 528 542 549
 551 560 563 566 566 577 579 607 615 615 623 624 633 643 649 653 665 690
 694 695 709 728 740
Program ended with exit code: 0
```