The Big-O time complexity of the delMax() is O(logn). for the swap and delete last, It’s time complexity are all O(1), for the sink, in every loop the range of the array is cut down one half, the time complexity is O(logn)., finally the total Big-O time complexity of the delMax() is O(logn).

```
while (leftChild(k) < size )
{
    int i = findMax(k);
    if(data[i] > data[k])
    {
        swap(k, i);
        k = i;
    }else
    {
        break;
    }
}
```

The Big-O time complexity of the daryHeapsort() is O(nlogn),

```
for(int i = 0; i < size; i++)
{
    sortedArray[i] = delMax();
}
```

As we can see the core code does a loop which have n time, in each loop, it call the delMax(), as we discussed just now, the Big-O time complexity for delMax() is O(logn), so the final Big-O time complexity of the daryHeapsort() is O(nlogn).