

Power Set Algorithm

powerSet(A[], R[])

③ [n = A.length, i = 0, max = 0; | + | + |

① - [float[] temp = new float[n];

① - temp[k] = A[0];

for (j = 0; j < n; j++) {

for (k = 1; k < n; k++) {

// add all values bigger than A[i] to temp

if (temp[i] <= A[k])

i++;
temp[i] = A[k];

$\sum_{k=1}^{n-1} 3$
 \Downarrow
 $\sum_{k=\emptyset}^{n-1} 3$

$\sum_{j=0}^{n-1} 3n + \sum_{j=0}^{n-1} 3$

$= 3n(n) + 3(n)$

$3 * (n-1 - 1 + 1) = 3n-3$

if (max < (i+1)) {

max(n+2)+2

System.arraycopy(temp, 0, R, n);

max = (i+1); | + |

}

return max

$3 + 1 + 1 + 3n^2 + 3n + n + 4 + 1$

PowerSet = $3n^2 + 4n + 10$

Power Set Algorithm

① $\text{int max} = 1$

① $\text{float } [] \text{ bestSubSq} = \text{new float } []$;

$(3n^2 + 4n + 1)$ $\text{max} = \text{powerSet}(A, \text{bestSubSq});$
 $1 + 3n^2 + 4n + 10$

Let $\text{max} = m$

// Copy bestSubSq to R

① $R = \text{new float } [\text{max}]$

$(m+1)$ $\left[\begin{array}{l} \text{for (int } i = 0; i < \text{max}; i++) \\ R[i] = \text{bestSubSq}[i]; \end{array} \right] \sum_{i=0}^m 1 = (m - 0 + 1) = m + 1$

$$1 + 1 + 3n^2 + 4n + 11 + 1 + m + 1$$

$$3n^2 + 4n + 15 + m$$

$$3n^2 + 4n + 15 + m \in O(n^2 + m)$$

Where n number of values in original input

Where m number of the longest non-decreasing subsequence in $A[]$

$$\lim_{\substack{n \rightarrow \infty \\ m \rightarrow \infty}} 3n^2 + 4n + 15 + m = \lim_{\substack{n \rightarrow \infty \\ m \rightarrow \infty}} \frac{3n^2 + 4n + 15}{n^2} + \lim_{m \rightarrow \infty} \frac{m}{m} = 3 + 1 > 0 \text{ and a constant}$$

therefore $3n^2 + 4n + 15 \in O(n^2)$ and $m \in O(m)$

then $O(n^2) + O(m) = O(n^2 + m)$

Power set algorithm

```
Problems @ Javadoc Declaration Console
<terminated> PowerSet [Java Application] C:\Program Files\Java\jre1.8.0_101\bin\javaw.exe (Oct 4, 2016, 8:44:27 PM)
CPSC 335-x - Programming Assignment #2
Longest non-decreasing subsequence problem, powerset algorithm
Enter the number of elements in the sequence: 5
Enter the elements in the sequence:
5 42 2 75 555
Input sequence
5 42 2 75 555
The longest non-decreasing subsequence has length 4The longest non-decreasing subsequence is
5 42 75 555
elapsed time: 0.016795 seconds
```

```
Problems @ Javadoc Declaration Console
<terminated> PowerSet [Java Application] C:\Program Files\Java\jre1.8.0_101\bin\javaw.exe (Oct 4, 2016, 8:45:39 PM)
CPSC 335-x - Programming Assignment #2
Longest non-decreasing subsequence problem, powerset algorithm
Enter the number of elements in the sequence: 10
Enter the elements in the sequence:
55 4 2 8 557 425 145 847 1204 10000
Input sequence
55 4 2 8 557 425 145 847 1204 10000
The longest non-decreasing subsequence has length 5The longest non-decreasing subsequence is
55 557 847 1204 10000
elapsed time: 0.02006 seconds
```

```
Problems @ Javadoc Declaration Console
<terminated> PowerSet [Java Application] C:\Program Files\Java\jre1.8.0_101\bin\javaw.exe (Oct 4, 2016, 8:46:59 PM)
CPSC 335-x - Programming Assignment #2
Longest non-decreasing subsequence problem, powerset algorithm
Enter the number of elements in the sequence: 8
Enter the elements in the sequence:
8 7 6 5 4 3 2 1
Input sequence
8 7 6 5 4 3 2 1
The longest non-decreasing subsequence has length 1The longest non-decreasing subsequence is
8
elapsed time: 0.019594 seconds
```