Time Complexity

Fn: remove Duplicates:

```
for (int i=0; i < arraySize-1; i++)           ∑(i=0 to n-1) 1
{ if (...)
  {  tempVar[i++] = arrg[i];}}                 ∑(i=0 to n-2) 1    ∑(i=0 to n-2) 1
```

$$\sum_{i=0}^{n-1} 1$$
$$\sum_{i=0}^{n-2} 1 \qquad \sum_{i=0}^{n-2} 1$$

~~= n-1 + n-2 + n-2~~

```
for (int i=0; i<j; i++)                        ∑(i=0 to j) 1
{ ... }                                        ∑(i=0 to j-1) 1
```

$$\sum_{i=0}^{j} 1$$
$$\sum_{i=0}^{j-1} 1$$

time: $n-1 + n-2 + n-2 + j + j - 1$

main:

```
for (int i=0; i < arraySize; i++)              ∑(i=0 to n) 1
{ ... }                                        ∑(i=0 to n-1) 1
```

$$\sum_{i=0}^{n} 1$$
$$\sum_{i=0}^{n-1} 1$$

```
for (int i=0; i < arraySize; i++)              ∑(i=0 to n) 1
{ ... }                                        ∑(i=0 to n-1) 1
```

$$\sum_{i=0}^{n} 1$$
$$\sum_{i=0}^{n-1} 1$$

time: $n + n-1 + n + n-1$

total: $n-1 + n-2 + n-2 + j + j - 1 + n + n-1 + n + n-1$
$$= O(n)$$

∅ Optimize: {in-place removal-so modify original array in-place.}
parallel processing