Problem 1

Time Complexity

```
for( int i=0; i<k; i++)
{  arrays[i] = malloc(sizeof(int).n);
   printf("...");
   for (int j=0; j<n; j++)
   {  scanf(...);}  }
```

$\sum_{i=0}^{k} 1$

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

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

$\sum_{i=0}^{n-1} \left(\sum_{j=0}^{n} 1 \right)$

$\sum_{i=0}^{k-1} \left(\sum_{j=0}^{n-1} 1 \right)$

$= k + k-1 + k-1 + k(k-1) + (k-1)(n-1)$

Merging arrays

```
for (int i=0; i<k; i++)
  {  for (int j=0; j<n; j++)
     {  merged Array[i*n+j] = arrays[i][j];}}
```

$\sum_{i=0}^{k} 1$

$\sum_{i=0}^{k-1} \left(\sum_{j=0}^{n} (1) \right)$

$\sum_{i=0}^{k-1} \left(\sum_{j=0}^{n-1} (1) \right)$

$= k + n(k-1) + (k-1)(n-1)$

Freeing memory:

```
for( int i=0; i<k; i++)
{  free( arrays[i]);}
```

$\sum_{i=0}^{k} (1)$

$= k$

$+$ Total $= k + k-1 + k-1 + n(k-1) + (k-1)(n-1) + k + n(k-1) + (k-1)(n-1)$

$= O(nk)$

optimize: 1. change sorting algorithm to quick sort.

2. instead of allocating mem. for each array, merge into 1 before allocating