EECS2040 Data Structure Hw #6 (Chapter 7 Sorting, Chapter 8 Hashing)
due date 6/27/2021,by 108061217 鍾永桓

Part 2 Coding

You should submit:
(a) All your source codes (C++ file).
(b) Show the execution trace of your program.

1. (50%) Sorting:
   Write a C++ program to perform 5 different sorting , insertion sort, median-of-three quick sort, iterative merge sort, recursive merge sort, and heap sort, on lists of characters, integer, floating point numbers, and C++ strings.
   a. You need to write the 5 sorting function templates (refer to example programs in textbook or pptx)
   b. Randomly generate a list of 20 characters as an input unsorted list.
   c. Randomly generate a list of 20 integers as an input unsorted list.
   d. Randomly generate a list of 20 floats as an input unsorted list.
   e. Randomly generate a list of 20 string objects as an input unsorted list.

   Show your results using the above 3 lists in your program.

   ANS:

List of characters:

```
InsertionSort:
I,X,W,p,l,h,O,s,p,O,P,I,w,P,N,v,H,X,w,B,
I,X,W,p,l,h,O,s,p,O,P,I,w,P,N,v,H,X,w,B,
I,W,X,p,l,h,O,s,p,O,P,I,w,P,N,v,H,X,w,B,
I,W,X,p,l,h,O,s,p,O,P,I,w,P,N,v,H,X,w,B,
I,W,X,l,p,h,O,s,p,O,P,I,w,P,N,v,H,X,w,B,
I,W,X,h,l,p,O,s,p,O,P,I,w,P,N,v,H,X,w,B,
I,O,W,X,h,l,p,s,p,O,P,I,w,P,N,v,H,X,w,B,
I,O,W,X,h,l,p,s,p,O,P,I,w,P,N,v,H,X,w,B,
I,O,W,X,h,l,p,p,s,O,P,I,w,P,N,v,H,X,w,B,
I,O,O,W,X,h,l,p,p,s,P,I,w,P,N,v,H,X,w,B,
I,O,O,P,W,X,h,l,p,p,s,I,w,P,N,v,H,X,w,B,
I,I,O,O,P,W,X,h,l,p,p,s,w,P,N,v,H,X,w,B,
I,I,O,O,P,W,X,h,l,p,p,s,w,P,N,v,H,X,w,B,
I,I,O,O,P,P,W,X,h,l,p,p,s,w,N,v,H,X,w,B,
I,I,N,O,O,P,P,W,X,h,l,p,p,s,w,v,H,X,w,B,
I,I,N,O,O,P,P,W,X,h,l,p,p,s,v,w,H,X,w,B,
H,I,I,N,O,O,P,P,W,X,h,l,p,p,s,v,w,X,w,B,
H,I,I,N,O,O,P,P,W,X,X,h,l,p,p,s,v,w,w,B,
H,I,I,N,O,O,P,P,W,X,X,h,l,p,p,s,v,w,w,B,
B,H,I,I,N,O,O,P,P,W,X,X,h,l,p,p,s,v,w,w,
final list
B,H,I,I,N,O,O,P,P,W,X,X,h,l,p,p,s,v,w,w,

generate another list
QuickSort:
l,q,X,x,l,y,v,X,i,Y,m,S,Y,k,k,J,R,a,B,i,
l,i,X,x,l,y,v,X,i,Y,m,S,Y,k,k,J,R,a,B,q,
l,i,X,B,l,y,v,X,i,Y,m,S,Y,k,k,J,R,a,x,q,
l,i,X,B,a,y,v,X,i,Y,m,S,Y,k,k,J,R,l,x,q,
l,i,X,B,a,R,v,X,i,Y,m,S,Y,k,k,J,y,l,x,q,
l,i,X,B,a,R,J,X,i,Y,m,S,Y,k,k,v,y,l,x,q,
l,i,X,B,a,R,J,X,i,Y,k,S,Y,k,m,v,y,l,x,q,
k,i,X,B,a,R,J,X,i,Y,k,S,Y,l,m,v,y,l,x,q,
k,i,X,B,a,R,J,X,i,Y,Y,S,k,l,m,v,y,l,x,q,
S,i,X,B,a,R,J,X,i,Y,Y,k,k,l,m,v,y,l,x,q,
S,J,X,B,a,R,i,X,i,Y,Y,k,k,l,m,v,y,l,x,q,
S,J,R,B,a,X,i,X,i,Y,Y,k,k,l,m,v,y,l,x,q,
B,J,R,S,a,X,i,X,i,Y,Y,k,k,l,m,v,y,l,x,q,
B,J,R,S,a,X,i,X,i,Y,Y,k,k,l,m,v,y,l,x,q,
B,J,R,S,a,X,i,X,i,Y,Y,k,k,l,m,v,y,l,x,q,
B,J,R,S,a,X,X,X,i,Y,i,k,k,l,m,v,y,l,x,q,
B,J,R,S,a,X,Y,X,Y,i,i,k,k,l,m,v,y,l,x,q,
B,J,R,S,Y,X,Y,X,a,i,i,k,k,l,m,v,y,l,x,q,
B,J,R,S,Y,X,X,Y,a,i,i,k,k,l,m,v,y,l,x,q,
B,J,R,S,X,X,Y,Y,a,i,i,k,k,l,m,v,y,l,x,q,
B,J,R,S,X,X,Y,Y,a,i,i,k,k,l,m,v,y,l,x,q,
B,J,R,S,X,X,Y,Y,a,i,i,k,k,l,m,v,y,l,x,q,
B,J,R,S,X,X,Y,Y,a,i,i,k,k,l,m,l,y,v,x,q,
B,J,R,S,X,X,Y,Y,a,i,i,k,k,l,l,m,y,v,x,q,
B,J,R,S,X,X,Y,Y,a,i,i,k,k,l,l,m,q,v,x,y,
B,J,R,S,X,X,Y,Y,a,i,i,k,k,l,l,m,q,v,x,y,
final list
B,J,R,S,X,X,Y,Y,a,i,i,k,k,l,l,m,q,v,x,y,
```

```
generate another list
iterative merge sort:
o,s,q,l,k,E,g,T,D,E,U,o,u,r,N,I,S,J,f,A,
o,s,l,q,E,k,T,g,D,E,U,o,r,u,I,N,J,S,A,f,
l,o,q,s,E,T,g,k,D,E,U,o,I,N,r,u,A,J,S,f,
E,T,g,k,l,o,q,s,D,E,I,N,U,o,r,u,A,J,S,f,
D,E,E,I,N,T,U,g,k,l,o,o,q,r,s,u,A,J,S,f,
A,D,E,E,I,J,N,S,T,U,f,g,k,l,o,o,q,r,s,u,
final list
A,D,E,E,I,J,N,S,T,U,f,g,k,l,o,o,q,r,s,u,

generate another list

recursive merge sort:
link:
6,18,3,10,11,15,4,12,16,5,19,20,14,7,0,1,17,9,13,8,
final list
D,F,G,H,R,T,U,U,X,Y,c,d,e,f,h,n,o,q,u,w,

generate another list

HeapSort:
y,y,v,w,r,u,t,o,u,C,g,D,l,W,J,Y,O,b,Y,B,
y,w,v,u,r,u,t,o,b,C,g,D,l,W,J,Y,O,B,Y,y,
w,u,v,o,r,u,t,Y,b,C,g,D,l,W,J,Y,O,B,y,y,
v,u,u,o,r,l,t,Y,b,C,g,D,B,W,J,Y,O,w,y,y,
u,r,u,o,g,l,t,Y,b,C,O,D,B,W,J,Y,v,w,y,y,
u,r,t,o,g,l,Y,Y,b,C,O,D,B,W,J,u,v,w,y,y,
t,r,l,o,g,J,Y,Y,b,C,O,D,B,W,u,u,v,w,y,y,
r,o,l,b,g,J,Y,Y,W,C,O,D,B,t,u,u,v,w,y,y,
o,g,l,b,O,J,Y,Y,W,C,B,D,r,t,u,u,v,w,y,y,
l,g,Y,b,O,J,D,Y,W,C,B,o,r,t,u,u,v,w,y,y,
g,b,Y,Y,O,J,D,B,W,C,l,o,r,t,u,u,v,w,y,y,
b,Y,Y,W,O,J,D,B,C,g,l,o,r,t,u,u,v,w,y,y,
Y,W,Y,C,O,J,D,B,b,g,l,o,r,t,u,u,v,w,y,y,
Y,W,J,C,O,B,D,Y,b,g,l,o,r,t,u,u,v,w,y,y,
W,O,J,C,D,B,Y,Y,b,g,l,o,r,t,u,u,v,w,y,y,
O,D,J,C,B,W,Y,Y,b,g,l,o,r,t,u,u,v,w,y,y,
J,D,B,C,O,W,Y,Y,b,g,l,o,r,t,u,u,v,w,y,y,
D,C,B,J,O,W,Y,Y,b,g,l,o,r,t,u,u,v,w,y,y,
C,B,D,J,O,W,Y,Y,b,g,l,o,r,t,u,u,v,w,y,y,
final list
B,C,D,J,O,W,Y,Y,b,g,l,o,r,t,u,u,v,w,y,y,
```

List of integers:

```
InsertionSort:
374,197,312,267,364,46,132,404,118,487,490,159,181,183,145,261,424,421,2,392,
197,374,312,267,364,46,132,404,118,487,490,159,181,183,145,261,424,421,2,392,
197,312,374,267,364,46,132,404,118,487,490,159,181,183,145,261,424,421,2,392,
197,267,312,374,364,46,132,404,118,487,490,159,181,183,145,261,424,421,2,392,
197,267,312,364,374,46,132,404,118,487,490,159,181,183,145,261,424,421,2,392,
46,197,267,312,364,374,132,404,118,487,490,159,181,183,145,261,424,421,2,392,
46,132,197,267,312,364,374,404,118,487,490,159,181,183,145,261,424,421,2,392,
46,132,197,267,312,364,374,404,118,487,490,159,181,183,145,261,424,421,2,392,
46,118,132,197,267,312,364,374,404,487,490,159,181,183,145,261,424,421,2,392,
46,118,132,197,267,312,364,374,404,487,490,159,181,183,145,261,424,421,2,392,
46,118,132,197,267,312,364,374,404,487,490,159,181,183,145,261,424,421,2,392,
46,118,132,159,197,267,312,364,374,404,487,490,181,183,145,261,424,421,2,392,
46,118,132,159,181,197,267,312,364,374,404,487,490,183,145,261,424,421,2,392,
46,118,132,159,181,183,197,267,312,364,374,404,487,490,145,261,424,421,2,392,
46,118,132,145,159,181,183,197,267,312,364,374,404,487,490,261,424,421,2,392,
46,118,132,145,159,181,183,197,261,267,312,364,374,404,487,490,424,421,2,392,
46,118,132,145,159,181,183,197,261,267,312,364,374,404,424,487,490,421,2,392,
46,118,132,145,159,181,183,197,261,267,312,364,374,404,421,424,487,490,2,392,
2,46,118,132,145,159,181,183,197,261,267,312,364,374,404,421,424,487,490,392,
2,46,118,132,145,159,181,183,197,261,267,312,364,374,392,404,421,424,487,490,
final list
2,46,118,132,145,159,181,183,197,261,267,312,364,374,392,404,421,424,487,490,

generate another list
QuickSort:
212,490,192,0,205,129,422,93,305,295,296,31,344,108,151,209,154,283,113,124,
212,124,192,0,205,129,422,93,305,295,296,31,344,108,151,209,154,283,113,490,
212,124,192,0,205,129,113,93,305,295,296,31,344,108,151,209,154,283,422,490,
212,124,192,0,205,129,113,93,154,295,296,31,344,108,151,209,305,283,422,490,
212,124,192,0,205,129,113,93,154,209,296,31,344,108,151,295,305,283,422,490,
212,124,192,0,205,129,113,93,154,209,151,31,344,108,296,295,305,283,422,490,
212,124,192,0,205,129,113,93,154,209,151,31,108,344,296,295,305,283,422,490,
108,124,192,0,205,129,113,93,154,209,151,31,212,344,296,295,305,283,422,490,
108,31,192,0,205,129,113,93,154,209,151,124,212,344,296,295,305,283,422,490,
108,31,93,0,205,129,113,192,154,209,151,124,212,344,296,295,305,283,422,490,
0,31,93,108,205,129,113,192,154,209,151,124,212,344,296,295,305,283,422,490,
0,31,93,108,205,129,113,192,154,209,151,124,212,344,296,295,305,283,422,490,
0,31,93,108,205,129,113,192,154,209,151,124,212,344,296,295,305,283,422,490,
0,31,93,108,205,129,113,192,154,124,151,209,212,344,296,295,305,283,422,490,
0,31,93,108,151,129,113,192,154,124,205,209,212,344,296,295,305,283,422,490,
0,31,93,108,151,129,113,124,154,192,205,209,212,344,296,295,305,283,422,490,
0,31,93,108,124,129,113,151,154,192,205,209,212,344,296,295,305,283,422,490,
0,31,93,108,124,113,129,151,154,192,205,209,212,344,296,295,305,283,422,490,
0,31,93,108,113,124,129,151,154,192,205,209,212,344,296,295,305,283,422,490,
0,31,93,108,113,124,129,151,154,192,205,209,212,344,296,295,305,283,422,490,
0,31,93,108,113,124,129,151,154,192,205,209,212,283,296,295,305,344,422,490,
0,31,93,108,113,124,129,151,154,192,205,209,212,283,296,295,305,344,422,490,
0,31,93,108,113,124,129,151,154,192,205,209,212,283,295,296,305,344,422,490,
final list
0,31,93,108,113,124,129,151,154,192,205,209,212,283,295,296,305,344,422,490,
```

```
generate another list
iterative merge sort:
122,103,283,155,138,280,416,415,201,418,159,413,409,351,266,466,332,188,411,489
103,122,155,283,138,280,415,416,201,418,159,413,351,409,266,466,188,332,411,489
103,122,155,283,138,280,415,416,159,201,413,418,266,351,409,466,188,332,411,489
103,122,138,155,280,283,415,416,159,201,266,351,409,413,418,466,188,332,411,489
103,122,138,155,159,201,266,280,283,351,409,413,415,416,418,466,188,332,411,489
103,122,138,155,159,188,201,266,280,283,332,351,409,411,413,415,416,418,466,489
final list
103,122,138,155,159,188,201,266,280,283,332,351,409,411,413,415,416,418,466,489

generate another list
483 207 20 328 316 171 389 322 306 354 447 428 309 82 83 448 215 351 215 416
recursive merge sort:
link:
3,0,17,14,18,8,2,20,4,13,7,16,11,5,15,6,1,19,10,9,
final list
20,82,83,171,207,215,215,306,309,316,322,328,351,354,389,416,428,447,448,483,

generate another list
270 374 330 31 77 448 349 409 136 260 398 472 320 418 152 488 442 41 162 248
HeapSort:
488,442,472,409,398,448,418,374,162,260,77,330,320,349,152,270,31,41,136,248,
472,442,448,409,398,330,418,374,162,260,77,248,320,349,152,270,31,41,136,488,
448,442,418,409,398,330,349,374,162,260,77,248,320,136,152,270,31,41,472,488,
442,409,418,374,398,330,349,270,162,260,77,248,320,136,152,41,31,448,472,488,
418,409,349,374,398,330,152,270,162,260,77,248,320,136,31,41,442,448,472,488,
409,398,349,374,260,330,152,270,162,41,77,248,320,136,31,418,442,448,472,488,
398,374,349,270,260,330,152,31,162,41,77,248,320,136,409,418,442,448,472,488,
374,270,349,162,260,330,152,31,136,41,77,248,320,398,409,418,442,448,472,488,
349,270,330,162,260,320,152,31,136,41,77,248,374,398,409,418,442,448,472,488,
330,270,320,162,260,248,152,31,136,41,77,349,374,398,409,418,442,448,472,488,
320,270,248,162,260,77,152,31,136,41,330,349,374,398,409,418,442,448,472,488,
270,260,248,162,41,77,152,31,136,320,330,349,374,398,409,418,442,448,472,488,
260,162,248,136,41,77,152,31,270,320,330,349,374,398,409,418,442,448,472,488,
248,162,152,136,41,77,31,260,270,320,330,349,374,398,409,418,442,448,472,488,
162,136,152,31,41,77,248,260,270,320,330,349,374,398,409,418,442,448,472,488,
152,136,77,31,41,162,248,260,270,320,330,349,374,398,409,418,442,448,472,488,
136,41,77,31,152,162,248,260,270,320,330,349,374,398,409,418,442,448,472,488,
77,41,31,136,152,162,248,260,270,320,330,349,374,398,409,418,442,448,472,488,
41,31,77,136,152,162,248,260,270,320,330,349,374,398,409,418,442,448,472,488,
final list
31,41,77,136,152,162,248,260,270,320,330,349,374,398,409,418,442,448,472,488,
```

List of floating point numbers:

```
InsertionSort:
4.84,4.53,1.96,0.26,3.21,3.21,0.67,4.44,3.46,0.09,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
4.53,4.84,1.96,0.26,3.21,3.21,0.67,4.44,3.46,0.09,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
1.96,4.53,4.84,0.26,3.21,3.21,0.67,4.44,3.46,0.09,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.26,1.96,4.53,4.84,3.21,3.21,0.67,4.44,3.46,0.09,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.26,1.96,3.21,4.53,4.84,3.21,0.67,4.44,3.46,0.09,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.26,1.96,3.21,3.21,4.53,4.84,0.67,4.44,3.46,0.09,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.26,0.67,1.96,3.21,3.21,4.53,4.84,4.44,3.46,0.09,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.26,0.67,1.96,3.21,3.21,4.44,4.53,4.84,3.46,0.09,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.26,0.67,1.96,3.21,3.21,3.46,4.44,4.53,4.84,0.09,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.09,0.26,0.67,1.96,3.21,3.21,3.46,4.44,4.53,4.84,3.52,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.09,0.26,0.67,1.96,3.21,3.21,3.46,3.52,4.44,4.53,4.84,1.23,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.09,0.26,0.67,1.23,1.96,3.21,3.21,3.46,3.52,4.44,4.53,4.84,4.75,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.09,0.26,0.67,1.23,1.96,3.21,3.21,3.46,3.52,4.44,4.53,4.75,4.84,2.14,2.26,2.82,0.53,1.39,3.92,0.22,
0.09,0.26,0.67,1.23,1.96,2.14,3.21,3.21,3.46,3.52,4.44,4.53,4.75,4.84,2.26,2.82,0.53,1.39,3.92,0.22,
0.09,0.26,0.67,1.23,1.96,2.14,2.26,3.21,3.21,3.46,3.52,4.44,4.53,4.75,4.84,2.82,0.53,1.39,3.92,0.22,
0.09,0.26,0.67,1.23,1.96,2.14,2.26,2.82,3.21,3.21,3.46,3.52,4.44,4.53,4.75,4.84,0.53,1.39,3.92,0.22,
0.09,0.26,0.53,0.67,1.23,1.96,2.14,2.26,2.82,3.21,3.21,3.46,3.52,4.44,4.53,4.75,4.84,1.39,3.92,0.22,
0.09,0.26,0.53,0.67,1.23,1.39,1.96,2.14,2.26,2.82,3.21,3.21,3.46,3.52,4.44,4.53,4.75,4.84,3.92,0.22,
0.09,0.26,0.53,0.67,1.23,1.39,1.96,2.14,2.26,2.82,3.21,3.21,3.46,3.52,3.92,4.44,4.53,4.75,4.84,0.22,
0.09,0.22,0.26,0.53,0.67,1.23,1.39,1.96,2.14,2.26,2.82,3.21,3.21,3.46,3.52,3.92,4.44,4.53,4.75,4.84,
final list
0.09,0.22,0.26,0.53,0.67,1.23,1.39,1.96,2.14,2.26,2.82,3.21,3.21,3.46,3.52,3.92,4.44,4.53,4.75,4.84,

generate another list
QuickSort:
0.19,2.71,2,4.46,1.97,2.77,1.61,0.47,1.87,2.74,2.66,0.23,2.27,4.62,0.49,4.01,1.35,1.17,1.97,4.82,
0.19,2.71,2,4.46,1.97,2.77,1.61,0.47,1.87,2.74,2.66,0.23,2.27,4.62,0.49,4.01,1.35,1.17,1.97,4.82,
0.19,2.71,2,1.97,1.97,2.77,1.61,0.47,1.87,2.74,2.66,0.23,2.27,4.62,0.49,4.01,1.35,1.17,4.46,4.82,
0.19,2.71,2,1.97,1.97,1.17,1.61,0.47,1.87,2.74,2.66,0.23,2.27,4.62,0.49,4.01,1.35,2.77,4.46,4.82,
0.19,2.71,2,1.97,1.97,1.17,1.61,0.47,1.87,1.35,2.66,0.23,2.27,4.62,0.49,4.01,2.74,2.77,4.46,4.82,
0.19,2.71,2,1.97,1.97,1.17,1.61,0.47,1.87,1.35,2.66,0.23,2.27,0.49,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.49,2,1.97,1.97,1.17,1.61,0.47,1.87,1.35,2.66,0.23,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.49,0.23,1.97,1.97,1.17,1.61,0.47,1.87,1.35,2.66,2,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.49,0.23,0.47,1.97,1.17,1.61,1.97,1.87,1.35,2.66,2,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.47,0.23,0.49,1.97,1.17,1.61,1.97,1.87,1.35,2.66,2,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.23,0.47,0.49,1.97,1.17,1.61,1.97,1.87,1.35,2.66,2,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.23,0.47,0.49,1.97,1.17,1.61,1.35,1.87,1.97,2.66,2,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.23,0.47,0.49,1.87,1.17,1.61,1.35,1.97,1.97,2.66,2,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.23,0.47,0.49,1.35,1.17,1.61,1.87,1.97,1.97,2.66,2,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.23,0.47,0.49,1.17,1.35,1.61,1.87,1.97,1.97,2.66,2,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.23,0.47,0.49,1.17,1.35,1.61,1.87,1.97,1.97,2.66,2,2.27,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.23,0.47,0.49,1.17,1.35,1.61,1.87,1.97,1.97,2.27,2,2.66,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.23,0.47,0.49,1.17,1.35,1.61,1.87,1.97,1.97,2,2.27,2.66,2.71,4.62,4.01,2.74,2.77,4.46,4.82,
0.19,0.23,0.47,0.49,1.17,1.35,1.61,1.87,1.97,1.97,2,2.27,2.66,2.71,4.46,4.01,2.74,2.77,4.62,4.82,
0.19,0.23,0.47,0.49,1.17,1.35,1.61,1.87,1.97,1.97,2,2.27,2.66,2.71,2.77,4.01,2.74,4.46,4.62,4.82,
0.19,0.23,0.47,0.49,1.17,1.35,1.61,1.87,1.97,1.97,2,2.27,2.66,2.71,2.77,2.74,4.01,4.46,4.62,4.82,
final list
0.19,0.23,0.47,0.49,1.17,1.35,1.61,1.87,1.97,1.97,2,2.27,2.66,2.71,2.74,2.77,4.01,4.46,4.62,4.82,
```

```
generate another list
iterative merge sort:
4.78,0.49,1.05,4.53,1.15,1.84,2.35,1.69,3.23,1.27,1.91,1.94,2.5,2.43,1.4,4.47,0.21,1.53,3.46,0.6,
0.49,4.78,1.05,4.53,1.15,1.84,1.69,2.35,1.27,3.23,1.91,1.94,2.43,2.5,1.4,4.47,0.21,1.53,0.6,3.46,
0.49,1.05,4.53,4.78,1.15,1.69,1.84,2.35,1.27,1.91,1.94,3.23,1.4,2.43,2.5,4.47,0.21,0.6,1.53,3.46,
0.49,1.05,1.15,1.69,1.84,2.35,4.53,4.78,1.27,1.4,1.91,1.94,2.43,2.5,3.23,4.47,0.21,0.6,1.53,3.46,
0.49,1.05,1.15,1.27,1.4,1.69,1.84,1.91,1.94,2.35,2.43,2.5,3.23,4.47,4.53,4.78,0.21,0.6,1.53,3.46,
0.21,0.49,0.6,1.05,1.15,1.27,1.4,1.53,1.69,1.84,1.91,1.94,2.35,2.43,2.5,3.23,3.46,4.47,4.53,4.78,
final list
0.21,0.49,0.6,1.05,1.15,1.27,1.4,1.53,1.69,1.84,1.91,1.94,2.35,2.43,2.5,3.23,3.46,4.47,4.53,4.78,

generate another list

recursive merge sort:
link:
13,12,16,10,15,1,0,5,3,18,2,7,9,17,20,19,4,8,6,14,
final list
0.07,0.37,0.62,0.84,1.05,1.12,1.22,1.55,2.37,2.91,3.53,3.61,3.96,4.08,4.26,4.27,4.32,4.54,4.72,4.85,

generate another list

HeapSort:
4.51,4.51,4.07,3.34,4.3,3.1,2.99,2.85,2.99,2.39,3.63,1.29,0.48,0,1.36,2.37,0.53,0.44,2.01,1.98,
4.51,4.3,4.07,3.34,3.63,3.1,2.99,2.85,2.99,2.39,1.98,1.29,0.48,0,1.36,2.37,0.53,0.44,2.01,4.51,
4.3,3.63,4.07,3.34,2.39,3.1,2.99,2.85,2.99,2.01,1.98,1.29,0.48,0,1.36,2.37,0.53,0.44,4.51,4.51,
4.07,3.63,3.1,3.34,2.39,1.29,2.99,2.85,2.99,2.01,1.98,0.44,0.48,0,1.36,2.37,0.53,4.3,4.51,4.51,
3.63,3.34,3.1,2.99,2.39,1.29,2.99,2.85,0.53,2.01,1.98,0.44,0.48,0,1.36,2.37,4.07,4.3,4.51,4.51,
3.34,2.99,3.1,2.85,2.39,1.29,2.99,2.37,0.53,2.01,1.98,0.44,0.48,0,1.36,3.63,4.07,4.3,4.51,4.51,
3.1,2.99,2.99,2.85,2.39,1.29,1.36,2.37,0.53,2.01,1.98,0.44,0.48,0,3.34,3.63,4.07,4.3,4.51,4.51,
2.99,2.85,2.99,2.37,2.39,1.29,1.36,0,0.53,2.01,1.98,0.44,0.48,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
2.99,2.85,1.36,2.37,2.39,1.29,0.48,0,0.53,2.01,1.98,0.44,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
2.85,2.39,1.36,2.37,2.01,1.29,0.48,0,0.53,0.44,1.98,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
2.39,2.37,1.36,1.98,2.01,1.29,0.48,0,0.53,0.44,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
2.37,2.01,1.36,1.98,0.44,1.29,0.48,0,0.53,2.39,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
2.01,1.98,1.36,0.53,0.44,1.29,0.48,0,2.37,2.39,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
1.98,0.53,1.36,0,0.44,1.29,0.48,2.01,2.37,2.39,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
1.36,0.53,1.29,0,0.44,0.48,1.98,2.01,2.37,2.39,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
1.29,0.53,0.48,0,0.44,1.36,1.98,2.01,2.37,2.39,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
0.53,0.44,0.48,0,1.29,1.36,1.98,2.01,2.37,2.39,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
0.48,0.44,0,0.53,1.29,1.36,1.98,2.01,2.37,2.39,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
0.44,0,0.48,0.53,1.29,1.36,1.98,2.01,2.37,2.39,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
final list
0,0.44,0.48,0.53,1.29,1.36,1.98,2.01,2.37,2.39,2.85,2.99,2.99,3.1,3.34,3.63,4.07,4.3,4.51,4.51,
```

List of C++ strings:

```
InsertionSort:
ylA,Ey,rpE,VCi,LLX,kVL,KytnI,qDEM,RHv,gnbb,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,ylA,rpE,VCi,LLX,kVL,KytnI,qDEM,RHv,gnbb,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,rpE,ylA,VCi,LLX,kVL,KytnI,qDEM,RHv,gnbb,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,VCi,rpE,ylA,LLX,kVL,KytnI,qDEM,RHv,gnbb,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,LLX,VCi,rpE,ylA,kVL,KytnI,qDEM,RHv,gnbb,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,LLX,VCi,kVL,rpE,ylA,KytnI,qDEM,RHv,gnbb,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,KytnI,LLX,VCi,kVL,rpE,ylA,qDEM,RHv,gnbb,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,KytnI,LLX,VCi,kVL,qDEM,rpE,ylA,RHv,gnbb,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,KytnI,LLX,RHv,VCi,kVL,qDEM,rpE,ylA,gnbb,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,KytnI,LLX,RHv,VCi,gnbb,kVL,qDEM,rpE,ylA,qMZ,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,KytnI,LLX,RHv,VCi,gnbb,kVL,qDEM,qMZ,rpE,ylA,ISSZ,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,ISSZ,KytnI,LLX,RHv,VCi,gnbb,kVL,qDEM,qMZ,rpE,ylA,kNBM,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,ISSZ,KytnI,LLX,RHv,VCi,gnbb,kNBM,kVL,qDEM,qMZ,rpE,ylA,ogx,Rl,eaP,QEc,FV,lJR,LczE,
Ey,ISSZ,KytnI,LLX,RHv,VCi,gnbb,kNBM,kVL,ogx,qDEM,qMZ,rpE,ylA,Rl,eaP,QEc,FV,lJR,LczE,
Ey,ISSZ,KytnI,LLX,RHv,Rl,VCi,gnbb,kNBM,kVL,ogx,qDEM,qMZ,rpE,ylA,eaP,QEc,FV,lJR,LczE,
Ey,ISSZ,KytnI,LLX,RHv,Rl,VCi,eaP,gnbb,kNBM,kVL,ogx,qDEM,qMZ,rpE,ylA,QEc,FV,lJR,LczE,
Ey,ISSZ,KytnI,LLX,QEc,RHv,Rl,VCi,eaP,gnbb,kNBM,kVL,ogx,qDEM,qMZ,rpE,ylA,FV,lJR,LczE,
Ey,FV,ISSZ,KytnI,LLX,QEc,RHv,Rl,VCi,eaP,gnbb,kNBM,kVL,ogx,qDEM,qMZ,rpE,ylA,lJR,LczE,
Ey,FV,ISSZ,KytnI,LLX,QEc,RHv,Rl,VCi,eaP,gnbb,kNBM,kVL,lJR,ogx,qDEM,qMZ,rpE,ylA,LczE,
Ey,FV,ISSZ,KytnI,LLX,LczE,QEc,RHv,Rl,VCi,eaP,gnbb,kNBM,kVL,lJR,ogx,qDEM,qMZ,rpE,ylA,
final list
Ey,FV,ISSZ,KytnI,LLX,LczE,QEc,RHv,Rl,VCi,eaP,gnbb,kNBM,kVL,lJR,ogx,qDEM,qMZ,rpE,ylA,

generate another list

QuickSort:
wjD,HyW,zfONjF,nDp,CSLS,NVRk,NxOK,dC,HvA,yIxBU,VR,DwRQK,EBUj,NiQ,mm,GKhRwX,yJ,KV,lhX,ZLamLY,
wjD,HyW,ZLamLY,nDp,CSLS,NVRk,NxOK,dC,HvA,yIxBU,VR,DwRQK,EBUj,NiQ,mm,GKhRwX,yJ,KV,lhX,zfONjF,
wjD,HyW,ZLamLY,nDp,CSLS,NVRk,NxOK,dC,HvA,lhX,VR,DwRQK,EBUj,NiQ,mm,GKhRwX,yJ,KV,yIxBU,zfONjF,
wjD,HyW,ZLamLY,nDp,CSLS,NVRk,NxOK,dC,HvA,lhX,VR,DwRQK,EBUj,NiQ,mm,GKhRwX,KV,yJ,yIxBU,zfONjF,
KV,HyW,ZLamLY,nDp,CSLS,NVRk,NxOK,dC,HvA,lhX,VR,DwRQK,EBUj,NiQ,mm,GKhRwX,wjD,yJ,yIxBU,zfONjF,
KV,HyW,GKhRwX,nDp,CSLS,NVRk,NxOK,dC,HvA,lhX,VR,DwRQK,EBUj,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
KV,HyW,GKhRwX,EBUj,CSLS,NVRk,NxOK,dC,HvA,lhX,VR,DwRQK,nDp,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
KV,HyW,GKhRwX,EBUj,CSLS,DwRQK,NxOK,dC,HvA,lhX,VR,NVRk,nDp,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
KV,HyW,GKhRwX,EBUj,CSLS,DwRQK,HvA,dC,NxOK,lhX,VR,NVRk,nDp,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
HvA,HyW,GKhRwX,EBUj,CSLS,DwRQK,KV,dC,NxOK,lhX,VR,NVRk,nDp,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
HvA,DwRQK,GKhRwX,EBUj,CSLS,HyW,KV,dC,NxOK,lhX,VR,NVRk,nDp,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,GKhRwX,EBUj,HvA,HyW,KV,dC,NxOK,lhX,VR,NVRk,nDp,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,GKhRwX,EBUj,HvA,HyW,KV,dC,NxOK,lhX,VR,NVRk,nDp,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,GKhRwX,EBUj,HvA,HyW,KV,dC,NxOK,lhX,VR,NVRk,nDp,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,dC,NxOK,lhX,VR,NVRk,nDp,NiQ,mm,ZLamLY,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,dC,NxOK,ZLamLY,VR,NVRk,nDp,NiQ,mm,lhX,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,dC,NxOK,ZLamLY,VR,NVRk,NiQ,nDp,mm,lhX,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,NiQ,NxOK,ZLamLY,VR,NVRk,dC,nDp,mm,lhX,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,NiQ,NVRk,ZLamLY,VR,NxOK,dC,nDp,mm,lhX,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,NVRk,NiQ,ZLamLY,VR,NxOK,dC,nDp,mm,lhX,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,NVRk,NiQ,NxOK,VR,ZLamLY,dC,nDp,mm,lhX,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,NVRk,NiQ,NxOK,VR,ZLamLY,dC,nDp,mm,lhX,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,NVRk,NiQ,NxOK,VR,ZLamLY,dC,lhX,mm,nDp,wjD,yJ,yIxBU,zfONjF,
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,NVRk,NiQ,NxOK,VR,ZLamLY,dC,lhX,mm,nDp,wjD,yJ,yIxBU,zfONjF,
final list
CSLS,DwRQK,EBUj,GKhRwX,HvA,HyW,KV,NVRk,NiQ,NxOK,VR,ZLamLY,dC,lhX,mm,nDp,wjD,yIxBU,yJ,zfONjF,
```

```
generate another list

iterative merge sort:
GqtC,WC,BAs,EpY,ZFX,tdgH,npu,rW,XI,OAfB,hoKfS,kZd,rFb,lQcYOb,CYwMW,puiE,wb,sbD,nI,xRHx,
GqtC,WC,BAs,EpY,ZFX,tdgH,npu,rW,OAfB,XI,hoKfS,kZd,lQcYOb,rFb,CYwMW,puiE,sbD,wb,nI,xRHx,
BAs,EpY,GqtC,WC,ZFX,npu,rW,tdgH,OAfB,XI,hoKfS,kZd,CYwMW,lQcYOb,puiE,rFb,nI,sbD,wb,xRHx,
BAs,EpY,GqtC,WC,ZFX,npu,rW,tdgH,CYwMW,OAfB,XI,hoKfS,kZd,lQcYOb,puiE,rFb,nI,sbD,wb,xRHx,
BAs,CYwMW,EpY,GqtC,OAfB,WC,XI,ZFX,hoKfS,kZd,lQcYOb,npu,puiE,rFb,rW,tdgH,nI,sbD,wb,xRHx,
BAs,CYwMW,EpY,GqtC,OAfB,WC,XI,ZFX,hoKfS,kZd,lQcYOb,nI,npu,puiE,rFb,rW,sbD,tdgH,wb,xRHx,
final list
BAs,CYwMW,EpY,GqtC,OAfB,WC,XI,ZFX,hoKfS,kZd,lQcYOb,nI,npu,puiE,rFb,rW,sbD,tdgH,wb,xRHx,

generate another list

recursive merge sort:
link:
3,17,12,20,7,1,15,9,10,6,5,14,8,18,13,0,2,11,4,16,
final list
Gd,LPcTaG,MSB,Nw,THC,XsOseL,aUQY,ew,fNrR,gu,jhrG,oXu,qo,rFmK,tSWM,tcCY,uV,wNON,xQ,yZXd,

generate another list

HeapSort:
uHH,rzz,jO,pheJ,lpJy,aNYaD,gGJu,iex,fMKD,jmwds,Sns,II,AHy,MV,RE,Hu,PAB,bA,WkobGW,gGDh,
rzz,pheJ,jO,iex,lpJy,aNYaD,gGJu,gGDh,fMKD,jmwds,Sns,II,AHy,MV,RE,Hu,PAB,bA,WkobGW,uHH,
pheJ,lpJy,jO,iex,jmwds,aNYaD,gGJu,gGDh,fMKD,WkobGW,Sns,II,AHy,MV,RE,Hu,PAB,bA,rzz,uHH,
lpJy,jmwds,jO,iex,bA,aNYaD,gGJu,gGDh,fMKD,WkobGW,Sns,II,AHy,MV,RE,Hu,PAB,pheJ,rzz,uHH,
jmwds,iex,jO,gGDh,bA,aNYaD,gGJu,PAB,fMKD,WkobGW,Sns,II,AHy,MV,RE,Hu,lpJy,pheJ,rzz,uHH,
jO,iex,gGJu,gGDh,bA,aNYaD,RE,PAB,fMKD,WkobGW,Sns,II,AHy,MV,Hu,jmwds,lpJy,pheJ,rzz,uHH,
iex,gGDh,gGJu,fMKD,bA,aNYaD,RE,PAB,Hu,WkobGW,Sns,II,AHy,MV,jO,jmwds,lpJy,pheJ,rzz,uHH,
gGJu,gGDh,aNYaD,fMKD,bA,MV,RE,PAB,Hu,WkobGW,Sns,II,AHy,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
gGDh,fMKD,aNYaD,PAB,bA,MV,RE,AHy,Hu,WkobGW,Sns,II,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
fMKD,bA,aNYaD,PAB,WkobGW,MV,RE,AHy,Hu,II,Sns,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
bA,WkobGW,aNYaD,PAB,Sns,MV,RE,AHy,Hu,II,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
aNYaD,WkobGW,RE,PAB,Sns,MV,II,AHy,Hu,bA,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
WkobGW,Sns,RE,PAB,Hu,MV,II,AHy,aNYaD,bA,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
Sns,PAB,RE,AHy,Hu,MV,II,WkobGW,aNYaD,bA,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
RE,PAB,MV,AHy,Hu,II,Sns,WkobGW,aNYaD,bA,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
PAB,II,MV,AHy,Hu,RE,Sns,WkobGW,aNYaD,bA,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
MV,II,Hu,AHy,PAB,RE,Sns,WkobGW,aNYaD,bA,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
II,AHy,Hu,MV,PAB,RE,Sns,WkobGW,aNYaD,bA,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
Hu,AHy,II,MV,PAB,RE,Sns,WkobGW,aNYaD,bA,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
final list
AHy,Hu,II,MV,PAB,RE,Sns,WkobGW,aNYaD,bA,fMKD,gGDh,gGJu,iex,jO,jmwds,lpJy,pheJ,rzz,uHH,
```

I have used four kinds of lists(int , float , char , string). First, I randomly generate a list and then use one of the below sorting function. Then I generate another list and use another sorting function. I keep doing so in my demo until I use every kinds of lists to run every function below. These functions are used to sort list in different ways.

void InsertionSort(T*a,const int n)
void QuickSort(T *a, const int left, const int right)
void MergeSort(T *a, const int n)
int rMergeSort(T* a, int* link, const int left, const int right)
void HeapSort(T *a,const int n)

2. (50%) Hashing:

Write a C++ program to implement two simple symbol tables (dictionaries) using hash table with linear probing for collision and hash table with chaining. For simplicity,

a. Consider storing only the key (need not consider the (key, value) pair) in the symbol tables.
b. Furthermore, the key is a variable-length character array with the first character of the key is an alphabet, e.g., abc, abcde, b, bye, cool,…
c. Consider a simple hash function using only the first character of key to hash, so h(abcde) = h(abc), h(b) = h(bye),.., etc. Therefore, collision can happen frequently.
d. The initial hash table size can be set to 26 since we have 26 alphabets which are the hashed keys.

Create 2 symbol table classes for linear probing and chaining, respectively. Both must implement at least the following functions:

Constructor,
Insert(key)
Search(key)

You may add other functions needed in your program.
Your main function may contains code like:

SymbolTable1 d1;
Setup at least 10 key objects
Insert those 10 keys into d1.
Display d1
Demo the search function of d1 (try at least 5 keys)

SymbolTable2 d2;

Setup at least 10 key objects

Insert those 10 keys into d1.

Display d2

Demo the search function of d2(try at least 5 keys)

ANS:

linear probing:

```
Setup and Insert 12 keys objects:GA,D,A,G,L,A2,A1,A3,A4,Z,ZA,E
Display:
ht[0]: A
ht[1]: A2
ht[2]: A1
ht[3]: D
ht[4]: A3
ht[5]: A4
ht[6]: GA
ht[7]: G
ht[8]: ZA
ht[9]: E
ht[10]:
ht[11]: L
ht[12]:
ht[13]:
ht[14]:
ht[15]:
ht[16]:
ht[17]:
ht[18]:
ht[19]:
ht[20]:
ht[21]:
ht[22]:
ht[23]:
ht[24]:
ht[25]: Z
Search("GA"): address:0x555a0a5a5360, key: GA
Search("A"): address:0x555a0a5a53c0, key: A
Search("A2"): address:0x555a0a5a5450, key: A2
Search("ZA"): address:0x555a0a5a5540, key: ZA
Search("E"): address:0x555a0a5a5570, key: E
```

hash table with chaining:

```
Setup and Insert 12 keys objects:A4,D,E,G,L,ZA,A3,GA,Z,A1,A2,A
Display:
0->A4->A3->A1->A2->A
1
2
3->D
4->E
5
6->G->GA
7
8
9
10
11->L
12
13
14
15
16
17
18
19
20
21
22
23
24
25->ZA->Z
Search("A4"): address:0x555a0a5a5b60, key: A4
Search("G"): address:0x555a0a5a5bf0, key: G
Search("ZA"): address:0x555a0a5a5c50, key: ZA
Search("GA"): address:0x555a0a5a5cb0, key: GA
Search("A"): address:0x555a0a5a5d70, key: A
```

According to b. of the problem, the key is a variable-length character array with the first character of the key is an alphabet, so I use the key with type of string. And according to a. , I storing only the key. In both linearprobing and chaining, the function Insert(const& k) can insert the key to the tables and the function Search(const string&k) can get the address of the searched key.(k is the key that we want to search or insert). The function Dispaly() can show the table.