```
// IED-001 (Prof. Dr. Silvio do Lago Pereira)

// ----------------------------------------------------------------------------
// Exemplo 2
// ----------------------------------------------------------------------------

void troca(int v[], int i, int j) {
   int x = v[i];
   v[i] = v[j];
   v[j] = x;
}

void bsort(int v[], int n) {
   for(int i=1; i<=n; i++)
      for(int j=0; j<n-i; j++)
         if( v[j]>v[j+1] )
            troca(v,j,j+1);
}


// ----------------------------------------------------------------------------
// Exercicio 1
// ----------------------------------------------------------------------------

#include <stdio.h>
...
int main(void) {
   int v[10] = {83,31,91,46,27,20,96,25,96,80};
   bsort(v,10);
   exibe(v,10);
   return 0;
}


// ----------------------------------------------------------------------------
// Exemplo 3
// ----------------------------------------------------------------------------

void intercala(int v[], int p, int m, int u) {
   int *w = malloc((u-p+1)*sizeof(int));
   int i=p, j=m+1, k=0;
   while( i<=m && j<=u )
      w[k++] = (v[i]<v[j]) ? v[i++] : v[j++];
   while( i<=m ) w[k++] = v[i++];
   while( j<=u ) w[k++] = v[j++];
   for(k=0; k<=u-p; k++) v[p+k] = w[k];
   free(w);
}

// ----------------------------------------------------------------------------
// Exercicio 4
// ----------------------------------------------------------------------------

#include <stdio.h>
...
int main(void) {
   int v[8] = {31,48,60,80,19,27,52,75};
   intercala(v,0,3,7);
   exibe(v,8);
   int w[9] = {10,82,27,38,41,53,60,75,99};
   intercala(w,0,1,9);
   exibe(w,9);
   return 0;
}

// ----------------------------------------------------------------------------
// Exemplo 5
// ----------------------------------------------------------------------------
```

```
void ms(int v[], int p, int u) {
   if( p==u ) return;
   int m = (p+u)/2;
   ms(v,p,m);
   ms(v,m+1,u);
   intercala(v,p,m,u);
}

void msort(int v[], int n) {
   ms(v,0,n-1);
}


// ------------------------------------------------------------------------
// Exercicio 5
// ------------------------------------------------------------------------

#include <stdio.h>
...
int main(void) {
   int v[10] = {83,31,91,46,27,20,96,25,96,80};
   msort(v,10);
   exibe(v,10);
   return 0;
}


// ------------------------------------------------------------------------
// Exercicio 6
// ------------------------------------------------------------------------

void preenche(int v[], int n, int s) {
   srand(s); // definida em stdlib.h
   for(int i=0; i<n; i++) v[i] = rand()%1000;
}


// ------------------------------------------------------------------------
// Exercicio 7
// ------------------------------------------------------------------------

int main(void) {
   int v[1e5];
   double t, b, m;
   puts("     n bsort msort");
   for(int n=1e4; n<=1e5; n+=1e4) {
      preenche(v,n,1);
      t = clock();                     // definida em time.h
      bsort(v,n);
      b = (clock()-t)/CLOCKS_PER_SEC;  // tempo do bsort
      preenche(v,n,1);
      t = clock();
      msort(v,n);
      m = (clock()-t)/CLOCKS_PER_SEC;  // tempo do bsort
      printf("%6d %5.1f %5.1f\n",n,b,m);
   }
   return 0;
}


// ------------------------------------------------------------------------
// Exercicio 8
// ------------------------------------------------------------------------

int main(void) {
   // precisamos usar malloc para criar vetores muito grandes!
   int *v = malloc(1e8*sizeof(int));
   puts("        n msort");
```

```
   for(int n=1e7; n<=1e8; n+=1e7) {
      preenche(v,n,1);
      double t = clock();
      msort(v,n);
      double m = (clock()-t)/CLOCKS_PER_SEC;
      printf("%9d %5.1f\n",n,m);
   }
   free(v);
   return 0;
}


// ----------------------------------------------------------------------------
// Exemplo 6
// ----------------------------------------------------------------------------

int lsearch(int x, int v[], int n) {
   for(int i=0; i<n; i++)
      if( x == v[i] )
         return 1;
   return 0;
}


// ----------------------------------------------------------------------------
// Exercicio 9
// ----------------------------------------------------------------------------

#include <stdio.h>
...
int main(void) {
   int v[8] = {66,80,31,48,27,75,19,52};
   printf("27: %d\n", lsearch(27,v,8));
   printf("51: %d\n", lsearch(51,v,8));
   return 0;
}


// ----------------------------------------------------------------------------
// Exemplo 7
// ----------------------------------------------------------------------------

int bsearch(int x, int v[], int n) {
   int p = 0;
   int u = n-1;
   while( p<=u ) {
      int m = (p+u)/2;
      if( x==v[m] ) return 1;
      if( x<v[m] ) u = m-1;
      else p = m+1;
   }
   return 0;
}

// ----------------------------------------------------------------------------
// Exercicio 10
// ----------------------------------------------------------------------------

#include <stdio.h>
...
int main(void) {
   int v[8] = {19,27,31,48,52,66,75,80};
   printf("27: %d\n", bsearch(27,v,8));
   printf("51: %d\n", bsearch(51,v,8));
   return 0;
}
```