```
/*
 * ================================================================================
 * Compile with
 *   gcc s0.c -o cs0.c -lm
 * Run with
 *   ./cs0 file0 file1
 *           EXERCISE
 * What's this program doing?
 * What's the difference between phi0 and phi1?
 * Execute this program under gdb
 * Use a tool like 'top' to get memory consumption for different sizes of the input files
 * ================================================================================
 */

#include <stdio.h>
#include<stdlib.h>
#include<math.h>

typedef
struct point{
   double x;
   double y;
} point_t;

typedef
struct seq{
   const point_t *pt;
   struct seq * tail;
} seq_t;

#define nil (seq_t *) 0
#define nullp(ptr) (ptr == nil)

seq_t *
cons(const point_t *pt, seq_t *xs)
{
   seq_t * x = (seq_t *) malloc(sizeof(seq_t));

   x->pt = pt;
   x->tail = xs;
   return x;
}

double
dist(const point_t *pt0, const point_t *pt1)
{
   return sqrt(pow(pt0->x - pt1->x, 2.0) + pow(pt0->y-pt1->y, 2.0)); // compare two calls to p
ower
}


double
phi0(seq_t *s0, seq_t *s1)
{
   if (nullp(s0) || nullp(s1)) return 0;

   return dist(s0->pt, s1->pt) + phi0(s0->tail, s1->tail);
}

double
phi1(seq_t *s0, seq_t *s1)
{
   double sum;

   while (1) {
      if (nullp(s0) || nullp(s1)) return sum;
      sum += dist(s0->pt, s1->pt);
      s0 = s0->tail;
      s1 = s1->tail;
   }
}

seq_t *
readPoints(const char *name)
{
   FILE *in = fopen(name, "r");
   seq_t *s0 = nil;
```

```
   while (1) {
      point_t * pt = (point_t *) malloc(sizeof(point_t));

      int n0 = fscanf(in, "%lf %lf", &pt->x, &pt->y);

      if ( 2 != n0) return s0;

      s0 = cons(pt, s0);
   }
}


int
main(int argc, char *argv[])
{
   point_t *pt0, *pt1;
   seq_t *s0 = readPoints(argv[1]);
   seq_t *s1 = readPoints(argv[2]);

   printf("%lf\n", phi0(s0, s1));
   printf("%lf\n", phi1(s0, s1));

   return 0;
}
```