```c
/*
 * ==========================================================================================
 * What's the difference between 's0.c' and 's1.c'?
 * ==========================================================================================
 */

#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;
   }
}


int
main(int argc, char *argv[])
{
   point_t *pt0, *pt1;
   FILE *in0, *in1;
   double sum = 0;
   seq_t *s0 = nil;
   seq_t *s1 = nil;

   in0 = fopen(argv[1], "r");
   in1 = fopen(argv[2], "r");
```

```
   while (1) {
      point_t * pt0 = (point_t *) malloc(sizeof(point_t));
      point_t * pt1 = (point_t *) malloc(sizeof(point_t));

      int n0 = fscanf(in0, "%lf %lf", &pt0->x, &pt0->y);
      int n1 = fscanf(in1, "%lf %lf", &pt1->x, &pt1->y);

      if ( ! (2 == n0) && (2 == n1) ) {
         printf("%lf\n", phi0(s0,s1));
         printf("%lf\n", phi1(s0,s1));
         return 0;
      }
      s0 = cons(pt0, s0);
      s1 = cons(pt1, s1);
   }

   return 0;
}
```