A i pick a digital elev file
     assume anything e.g. 600x600
     create window
     read data line by line
     colour pixels very simple scheme

 ii read true size from file, make window correct size.
     let user select which file is drawn
     extract other text info, so you know where corners are

 iii allow user to type a lat and lon
     work out pixel position and draw big blob

 iv given two (lat, lon) select the smallest map tile covering both
     here http://rabbit.eng.miami.edu/geographical/bintiles/coverage.txt
     is a new helpful file. It lists all the map tiles along with the range of
     latitude and longitude that they cover.

 v calculate/choose nice colours to make it look good.

B i read alphaplaces file, print useful info for selected places

 ii create structs for each, put in suitable D.S.
      hash table

A+B let user enter two places
   draw them on the right map.

C i read locations.txt, store as structs in useful D.S.
      probably a vector

 ii read closests file, augment alphaplaces structs so that they
     indicate their closest location (i.e. intersection)

B+C combine so user enters place name, program finds closest location
    and displays identifying info to be checked.

D   i   read majorroads, convert to structs, print selected few

C+D   i   build map structure locations are nodes, majorroads are edges.

         ii   system select random start location, print it and menu of roads
                 follow. User selects road, follow it and repeat.

B+C+D   let user enter starting place and do the same from there

A+B+C+D   make that appear interactively on the map.

E.   i   Adapt your priority queue to hold location objects.
                     augment location object to hold best distance.

         ii   Find shortest path from one place to another.
                     can be text-based, given numeric location n umbers
                     for start and destination points

         iii   when a location is taken from prio queue, for each road
                 inspected, draw a line for it on the map.

         iv   Display result on map.

A+B+C+D+E
         i   Do the whole thing

         ii   Print nice readable driving instructions.