Block Buffer:

Array of Structs:

struct Record{

str zip\_code,  
 place\_name,  
 state,  
 county,  
 latitude,  
 longitude;

};

Record Block[4]; //creates an array called Block, made up of 4 Record structs

Private Methods:

1. Read\_Block\_from\_data( offset );
   1. Read\_record(); X4
      1. Read\_zip\_code();
      2. Read\_place\_name();
      3. Read\_state();
      4. Read\_county();
      5. Read\_latitude();
      6. Read\_longitude();
2. Manipulation Methods for Block Data Stored in Memory
   1. Add\_record();
   2. Delete\_record();
   3. Update\_record();
   4. Display\_record();
   5. Compact\_record(); //will push all records to leftmost positions
3. Write\_Block\_to\_data( offset );
   1. Write\_record(); X4
      1. Write\_zip\_code();
      2. Write\_place\_name();
      3. Write\_state();
      4. Write\_county();
      5. Write\_latitude();
      6. Write\_longitude();
4. Write\_Block\_to\_SS( offset = RBN );
   1. write primary key of each record

User Functions:

1. User\_Delete\_record( primary key of record );
   1. Binary Search Index\_File for correct block
   2. Once the block is found take the RBN of the block
   3. Use RBN to find the corresponding Block in the SS\_File
      1. RBN + Header\_Record\_Size
   4. Use RBN to find the corresponding Block in the data.txt file
      1. offset = Records\_Per\_Block \* ( RBN - 1 ) + Header\_Record\_Size + 1
   5. Read\_Block\_from\_data( offset );
   6. Find the desired record from the block.
   7. Delete\_record();
   8. Compact\_record();
   9. Write\_Block\_to\_data( offset );
   10. Write\_Block\_to\_SS( offset = RBN );
2. User\_Add\_record( record to add )
   1. Binary Search Index\_File for correct block
   2. Once the block is found take the RBN of the block
   3. Use RBN to find the corresponding Block in the SS\_File
      1. RBN + Header\_Record\_Size
   4. Use RBN to find the corresponding Block in the data.txt file
      1. offset = Records\_Per\_Block \* ( RBN - 1 ) + Header\_Record\_Size + 1
   5. Read\_Block\_from\_data( offset ); -> From data.txt
   6. If block is not full -> Add\_record();
      1. Write\_Block\_to\_data( offset );
   7. If block is full -> check neighboring blocks -> Add\_record();
      1. Write\_Block\_to\_data( offset );
   8. If all blocks are full -> append new block
      1. Write\_Block\_to\_data( offset = number of records );
   9. Write\_Block\_to\_SS( offset = RBN );