Here is a program that uses an iterative version of Selection Sort:

```
#include <fstream>
#include <iostream>
using namespace std;

void selectionSort( int* a, int n );
int findMin( int*a, int n, int from );
void swap( int i, int j, int* a );

int main ()
{
  ifstream fin( "int_data.txt" );
  if( fin.fail() ) return 1;

  int n;
  fin >> n;
  int *A = createArray( n );

  fillArray( A, n, fin );
  printArray( A, n, cout );
  selectionSort( A, n );
  printArray( A, n, cout );

  deleteArray( A );

  fin.close();

  return 0;
}

void selectionSort( int* a, int n )
{
  int index_of_swap, index_of_min;
  for( index_of_swap = 0; index_of_swap <= n-2; index_of_swap++ )
  {
    index_of_min = findMin( a, n, index_of_swap );
    swap( index_of_swap, index_of_min, a );
  }
}

int findMin( int*a, int n, int from )
{
  int smallest = a[from];
  int ismall = from;
  for( int i = from + 1; i < n; i++ )
    if( a[i] < smallest )
    {
      smallest = a[i];
      ismall = i;
    }
  return ismall;
}

void swap( int i, int j, int* a )
{
  int tmp = a[i];
  a[i] = a[j];
  a[j] = tmp;
}
```

IC210 HW Extra Credit Name _____

Replace the existing function body of selectionSort to make it ***recursive***.

```
void selectionSort( int* a, int n )
{
   // recursive definition here:




}
```

Test it using the following as file “int_data.txt”:

```
10
2 5 7 6 3 0 1 9 4 8
```

Turn in: (1) This sheet with your name filled in, to which is stapled
(2) A hardopy listing of main.cpp
(3) A screen snapshot of program output.