P7 – Sudoku GUI 2

3.1 Create the GUI

```
//draw spinbox grid
auto *gridlayout = new QGridLayout();

//increment row (y)
for(int i=0; i<9; i++) {

   //increment column (x)
   for(int j=0; j<9; j++) {

      QSpinBox *temp = new QSpinBox();
      temp->setValue(sud.getCell(i, j));

      gridlayout->addWidget(temp, i, j);

      //gridlayout->itemAt((9*i) + j)->setValue(sud.getCell(i,j));
   }
}
```

```
//setup menu
auto *openMenu = new QAction("Open", this);
auto *closeMenu = new QAction("Close", this);
auto *solveMenu = new QAction("Solve", this);

QMenu *file = menuBar()->addMenu("File");

file->addAction(openMenu);
file->addAction(closeMenu);
file->addAction(solveMenu);
```

3.2 Initialize the GUI

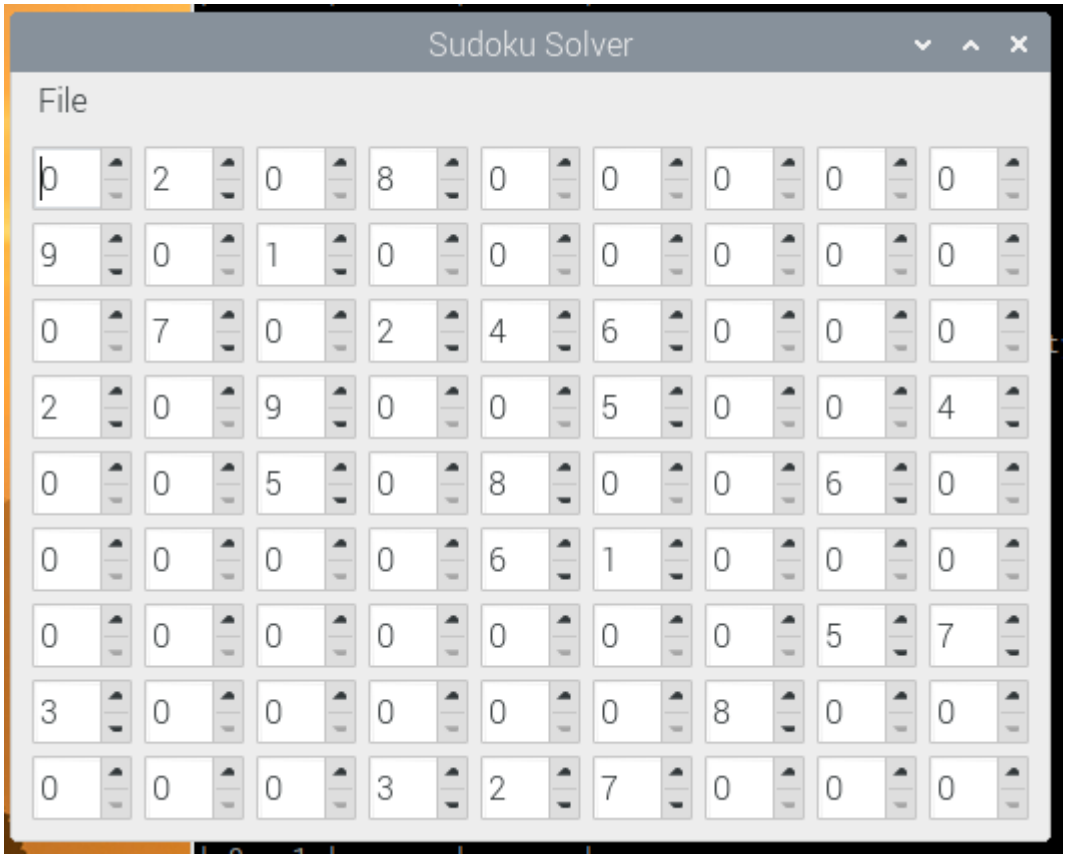
```
void Window::updateWidgets() {

   for(int i=0; i<9; i++) {

      for(int j=0; j<9; j++) {

         sudGrid[i][j]->setValue(sud.getCell(i, j));
      }
   }

}
```



3.3 Implement Signals and Slots

To accommodate signals and slots I had to store the widgets more permanently. To do this I just created a simple 2D array to store the SpinBox widgets so that their values can be changed later.

```
QSpinBox *sudGrid[9][9];

//draw spinbox grid
auto *gridlayout = new QGridLayout();

//increment row (y)
for(int i=0; i<9; i++) {

   //increment column (x)
   for(int j=0; j<9; j++) {

      sudGrid[i][j] = new QSpinBox();
      sudGrid[i][j]->setValue(sud.getCell(i, j));

      gridlayout->addWidget(sudGrid[i][j], i, j);
   }
}
```