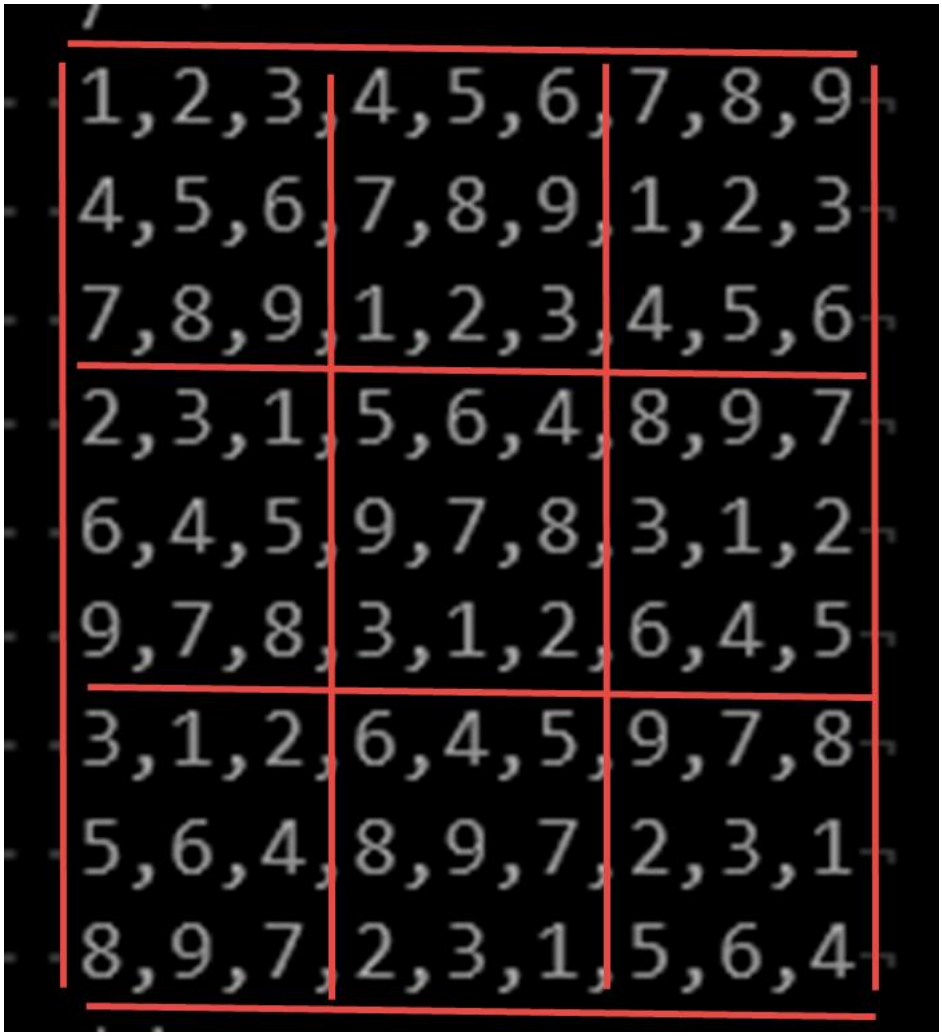
I have now had to take this down to the basics and create legitimate entry into the suduko permutations that will allow completion.
I will also need to use the smallest n objects in which r=9 can be nPr can be completed to meet objectives... This will be n=9

Below is an example of a grid that is permissible:



I have now created the equivalent entry that I would expect to see in the summary presented to end user...

1(0,0) 2(0,1) 3(0,2) 4(1,0) 5(1,1) 6(1,2) 7(2,0) 8(2,1) 9(2,2)

4(0,3) 5(0,4) 6(0,5) 7(1,3) 8(1,4) 9(1,5) 1(2,3) 2(2,4) 3(2,5)

7(0,6) 8(0,7) 9(0,8) 1(1,6) 2(1,7) 3(1,8) 4(2,6) 5(2,7) 6(2,8)

2(3,0) 3(3,1) 1(3,2) 6(4,0) 4(4,1) 5(4,2) 9(5,0) 7(5,1) 8(5,2)

5(3,3) 6(3,4) 4(3,5) 9(4,3) 7(4,4) 8(4,5) 3(5,3) 1(5,4) 2(5,5)

8(3,6) 9(3,7) 7(3,8) 3(4,6) 1(4,7) 2(4,8) 6(5,6) 4(5,7) 5(5,8)

3(6,0) 1(6,1) 2(6,2) 5(7,0) 6(7,1) 4(7,2) 8(8,0) 9(8,1) 7(8,2)

6(6,3) 4(6,4) 5(6,5) 8(7,3) 9(7,4) 7(7,5) 2(8,3) 3(8,4) 1(8,5)

9(6,6) 7(6,7) 8(6,8) 2(7,6) 3(7,7) 1(7,8) 5(8,6) 6(8,7) 4(8,8)

This is an illustrative example of content it will compare against (presented as part of the code).

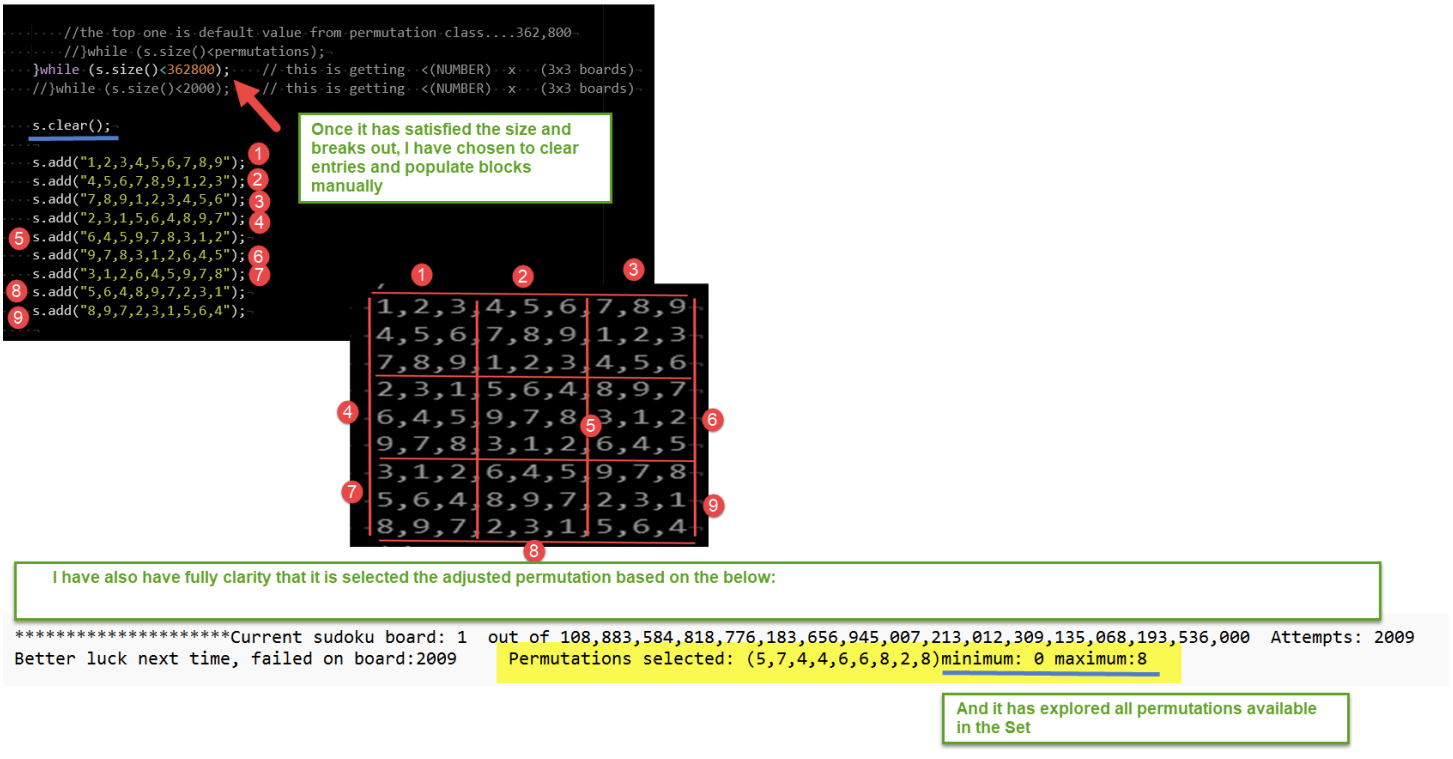
```
Moving onto Board Number: 1
*******************************************************************************

This is your board number 835 summary: 4(0,0) 5(0,1) 6(0,2) 7(1,0) 8(1,1) 9(1,2) 1(2,0) 2(2,1) 3(2,2) 6(0,3) 4(0,4) 5(0,5) 9(1,3) 7(1,4) 8(1,5) 3(2,3)
1(2,4) 2(2,5) 5(0,6) 6(0,7) 4(0,8) 8(1,6) 9(1,7) 7(1,8) 2(2,6) 3(2,7) 1(2,8) 2(3,0) 3(3,1) 1(3,2) 5(4,0) 6(4,1) 4(4,2) 8(5,0) 9(5,1) 7(5,2) 8(3,3) 9(3,4)
7(3,5) 2(4,3) 3(4,4) 1(4,5) 5(5,3) 6(5,4) 4(5,5) 1(3,6) 2(3,7) 3(3,8) 4(4,6) 5(4,7) 6(4,8) 7(5,6) 8(5,7) 9(5,8) 8(6,0) 9(6,1) 7(6,2) 2(7,0) 3(7,1) 1(7,2)
5(8,0) 6(8,1) 4(8,2) 9(6,3) 7(6,4) 8(6,5) 3(7,3) 1(7,4) 2(7,5) 6(8,3) 4(8,4) 5(8,5) 6(6,6) 4(6,7) 5(6,8) 9(7,6) 7(7,7) 8(7,8) 3(8,6) 1(8,7) 2(8,8)
*******************************************************************************
```

I have few issues now..
Firstly, the text in blue is 654 characters with white spaces....
Lots software applications begin to hit a limit.....

So I am downloading notepad++ and will perform a search for the exact string...

1(0,0) 2(0,1) 3(0,2) 4(1,0) 5(1,1) 6(1,2) 7(2,0) 8(2,1) 9(2,2) 4(0,3) 5(0,4) 6(0,5) 7(1,3) 8(1,4) 9(1,5) 1(2,3) 2(2,4) 3(2,5) 7(0,6) 8(0,7) 9(0,8) 1(1,6) 2(1,7) 3(1,8) 4(2,6) 5(2,7) 6(2,8) 2(3,0) 3(3,1) 1(3,2) 6(4,0) 4(4,1) 5(4,2) 9(5,0) 7(5,1) 8(5,2) 5(3,3) 6(3,4) 4(3,5) 9(4,3) 7(4,4) 8(4,5) 3(5,3) 1(5,4) 2(5,5) 8(3,6) 9(3,7) 7(3,8) 3(4,6) 1(4,7) 2(4,8) 6(5,6) 4(5,7) 5(5,8) 3(6,0) 1(6,1) 2(6,2) 5(7,0) 6(7,1) 4(7,2) 8(8,0) 9(8,1) 7(8,2) 6(6,3) 4(6,4) 5(6,5) 8(7,3) 9(7,4) 7(7,5) 2(8,3) 3(8,4) 1(8,5) 9(6,6) 7(6,7) 8(6,8) 2(7,6) 3(7,7) 1(7,8) 5(8,6) 6(8,7) 4(8,8)

The next question arise on how to populate a test sample into my existing code to provision 9 mini grids as above.
I have completed changes in this area of code.



I have also have fully clarity that it is selected the adjusted permutation based on the below:

```
**********Current sudoku board: 1  out of 108,883,584,818,776,183,656,945,007,213,012,309,135,068,193,536,000  Attempts: 2009
Better luck next time, failed on board:2009    Permutations selected: (5,7,4,4,6,6,8,2,8)minimum: 0 maximum:8
```

And it has explored all permutations available in the Set

I have had to create an extremely light version of my code with only a single output

I then examined the file which executed to over 2million attempts.

log.txt 22/08/2025 19:56 Text Document 1,344,732 KB

Exceeding 1GB in size

Over 2 million searches

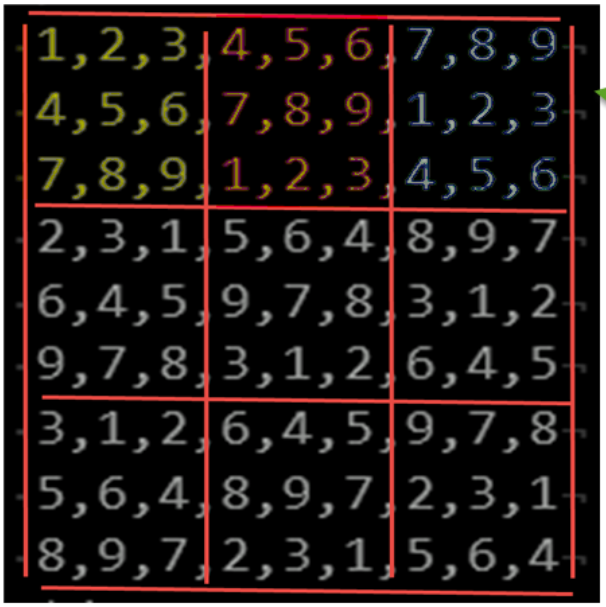
```
This is your board number 2039368 summary: 2(0,0) 3(0,1) 1(0,2) 5(1,0) 6(1,1) 4(1,2) 8(2,0) 9(2,1) 7(2,2) 9(0,3) 7(0,4) 8(0,5) 3(1,3) 1(1,4)
2(1,5) 6(2,3) 4(2,4) 5(2,5) 6(0,6) 4(0,7) 5(0,8) 9(1,6) 7(1,7) 8(1,8) 3(2,6) 1(2,7) 2(2,8) 1(3,0) 2(3,1) 3(3,2) 4(4,0) 5(4,1) 6(4,2) 7(5,0)
8(5,1) 9(5,2) 5(3,3) 6(3,4) 4(3,5) 8(4,3) 9(4,4) 7(4,5) 2(5,3) 3(5,4) 1(5,5) 1(3,6) 2(3,7) 3(3,8) 4(4,6) 5(4,7) 6(4,8) 7(5,6) 8(5,7) 9(5,8)
1(6,0) 2(6,1) 3(6,2) 4(7,0) 5(7,1) 6(7,2) 7(8,0) 8(8,1) 9(8,2) 4(6,3) 5(6,4) 6(6,5) 7(7,3) 8(7,4) 9(7,5) 1(8,3) 2(8,4) 3(8,5) 5(6,6) 6(6,7)
4(6,8) 8(7,6) 9(7,7) 7(7,8) 2(8,6) 3(8,7) 1(8,8)  Permutations selected: (4,6,2,1,7,8,0,5,3)minimum: 0 maximum:8
```

It did not reach any further than this through all efforts.

```
Permutations selected: (0,4,7,6,3,8,1,5,2)
```

As per code output

1(0,0) 2(0,1) 3(0,2) 4(1,0) 5(1,1) 6(1,2) 7(2,0) 8(2,1) 9(2,2) 4(0,3) 5(0,4) 6(0,5) 7(1,3) 8(1,4) 9(1,5) 1(2,3) 2(2,4) 3(2,5) 7(0,6) 8(0,7) 9(0,8) 1(1,6) 2(1,7) 3(1,8) 4(2,6) 5(2,7) 6(2,8) 2(3,0) 3(3,1) 1(3,2) 6(4,0) 4(4,1) 5(4,2) 9(5,0) 7(5,1) 8(5,2) 5(3,3) 6(3,4) 4(3,5) 9(4,3) 7(4,4) 8(4,5) 3(5,3) 1(5,4) 2(5,5) 8(3,6) 9(3,7) 7(3,8) 3(4,6) 1(4,7) 2(4,8) 6(5,6) 4(5,7) 5(5,8) 3(6,0) 1(6,1) 2(6,2) 5(7,0) 6(7,1) 4(7,2) 8(8,0) 9(8,1) 7(8,2) 6(6,3) 4(6,4) 5(6,5) 8(7,3) 9(7,4) 7(7,5) 2(8,3) 3(8,4) 1(8,5) 9(6,6) 7(6,7) 8(6,8) 2(7,6) 3(7,7) 1(7,8) 5(8,6) 6(8,7) 4(8,8)



This is furthest reach in 2 million entries.....

Also it is very confusing that the permutation random number is not corresponding to the same 3x3 block amongst all the board outputs...

For instance this is the output of code, it can be seen that

This is your board number 1 summary: 7(0,0) 8(0,1) 9(0,2) 1(1,0) 2(1,1) 3(1,2) 4(2,0) 5(2,1) 6(2,2) 7(0,3) 8(0,4) 9(0,5) 1(1,3) 2(1,4) 3(1,5) 4(2,3) 5(2,4) 6(2,5) 6(0,6) 4(0,7) 5(0,8) 9(1,6) 7(1,7) 8(1,8) 3(2,6) 1(2,7) 2(2,8) 5(3,0) 6(3,1) 4(3,2) 8(4,0) 9(4,1) 7(4,2) 2(5,0) 3(5,1) 1(5,2) 2(3,3) 3(3,4) 1(3,5) 5(4,3) 6(4,4) 4(4,5) 8(5,3) 9(5,4) 7(5,5) 3(3,6) 1(3,7) 2(3,8) 6(4,6) 4(4,7) 5(4,8) 9(5,6) 7(5,7) 8(5,8) 5(6,0) 6(6,1) 4(6,2) 8(7,0) 9(7,1) 7(7,2) 2(8,0) 3(8,1) 1(8,2) 5(6,3) 6(6,4) 4(6,5) 8(7,3) 9(7,4) 7(7,5) 2(8,3) 3(8,4) 1(8,5) 1(6,6) 2(6,7) 3(6,8) 4(7,6) 5(7,7) 6(7,8) 7(8,6) 8(8,7) 9(8,8) Permutations selected: (0,8,5,7,1,6,4,2,3)minimum: 0 maximum:8

This is your board number 5 summary: 9(0,0) 7(0,1) 8(0,2) 3(1,0) 1(1,1) 2(1,2) 6(2,0) 4(2,1) 5(2,2) 9(0,3) 7(0,4) 8(0,5) 3(1,3) 1(1,4) 2(1,5) 6(2,3) 4(2,4) 5(2,5) 2(0,6) 3(0,7) 1(0,8) 5(1,6) 6(1,7) 4(1,8) 8(2,6) 9(2,7) 7(2,8) 5(3,0) 6(3,1) 4(3,2) 8(4,0) 9(4,1) 7(4,2) 2(5,0) 3(5,1) 1(5,2) 9(3,3) 7(3,4) 8(3,5) 3(4,3) 1(4,4) 2(4,5) 6(5,3) 4(5,4) 5(5,5) 8(3,6) 9(3,7) 7(3,8) 2(4,6) 3(4,7) 1(4,8) 5(5,6) 6(5,7) 4(5,8) 9(6,0) 7(6,1) 8(6,2) 3(7,0) 1(7,1) 2(7,2) 6(8,0) 4(8,1) 5(8,2) 7(6,3) 8(6,4) 9(6,5) 1(7,3) 2(7,4) 3(7,5) 4(8,3) 5(8,4) 6(8,5) 2(6,6) 3(6,7) 1(6,8) 5(7,6) 6(7,7) 4(7,8) 8(8,6) 9(8,7) 7(8,8) Permutations selected: (0,2,3,4,8,7,1,5,6)minimum: 0 maximum:8

It makes little sense given that randomNumber is stored in a String array.

```
storeRetrieved3x3Grid[uniqueEntries]=randomNumber;
```

And then contents from the array are sent into StringJoiner:

```
permutationsSelected = new StringJoiner(",");
for (int z: storeRetrieved3x3Grid)
{

    //System.out.println(z);


    entry3x3=z;
    permutationsSelected.add(String.valueOf(entry3x3));
```

There are no references to the set in this process since all the values of the set are stored in the String already as part of:

```
String[] perm3x3 = s.toArray(new String[s.size()]);
```