Sudoku Players' Forums

Rating rules / Puzzles. Ordering the rules
Goto page **Previous** **1**, **2**, **3** ... , **19**, 20, **21** **Next**

newtopic **postreply**   **Sudoku Players' Forums Forum Index** -> **Advanced solving techniques**

Author	Message
coloin Joined: 05 May 2005 Posts: 1007 Location: Oxford	Posted: Mon Jul 06, 2009 4:03 am Post subject: quote **denis_berthier wrote:** Yes, but how many more different minimal puzzles shall we find in the second case? well the 30 clue puzzle is minimal......so none. all stats are minimal puzzles C
Back to top	profile pm
eleven Joined: 10 Feb 2008 Posts: 319	Posted: Mon Jul 06, 2009 4:26 am Post subject: quote **denis_berthier wrote:** But gcc doesn't compile it on my Mac (maybe I failed to put some necessary compilation options): The quick way to fix that is to make the string empty: char G[9999]=""; #include <stdio.h> ... PS: remove the last char, last line then m9:return solutions;} Last edited by eleven on Mon Jul 06, 2009 4:42 am; edited 1 time in total
Back to top	profile pm

denis_berthier
 	
Joined: 19 Jun 2007
Posts: 658
Location: Paris, France
	Posted: Mon Jul 06, 2009 4:41 am Post subject: quote
	
	eleven wrote:
	The fast way to fix that is to make the string empty:
	char G[9999]="";
	#include <stdio.h>
	...
	
	Better, but not enough
	
	Code:
	<pre>suexg14.c: In function 'main':
	suexg14.c:29: warning: incompatible implicit declaration of built-in function 'exit'
	suexg14.c:33: warning: incompatible implicit declaration of built-in function 'exit'
	suexg14.c:35: warning: incompatible implicit declaration of built-in function 'exit'
	suexg14.c:38: warning: incompatible implicit declaration of built-in function 'exit'
	suexg14.c:56: warning: incompatible implicit declaration of built-in function 'exit'
	suexg14.c: At top level:
	suexg14.c:109: error: stray '\26' in program</pre>

Back to top	profile pm www
eleven Joined: 10 Feb 2008 Posts: 319	Posted: Mon Jul 06, 2009 4:43 am Post subject: quote okay, replace all exit by return and remove the last line.
Back to top	profile pm
denis_berthier	Posted: Mon Jul 06, 2009 4:52 am Post subject: quote

Joined: 19 Jun 2007
Posts: 658
Location: Paris, France

> **coloin wrote:**
>> **denis_berthier wrote:**
>>> **coloin wrote:**
>>>> take a specific 20 clue puzzle and a 30 clue puzzle - both in this sub-grid
>>>> from the 40 clues
>>>> 20! ways to reach this 20 clue puzzle
>>>> 10! ways to reach this 30 clue puzzle
>>>
>>> Yes, but how many more different minimal puzzles shall we find in the second case?
>>
>> well the 30 clue puzzle is minimal......so none.
>> all stats are minimal puzzles

And is your 20-clue puzzle minimal also?
How many 20-clue minimal puzzles and how many 30-clue in your initial grid?

Back to top   [profile] [pm] [www]

---

denis_berthier   Posted: Mon Jul 06, 2009 4:54 am   Post subject: [quote]

Joined: 19 Jun 2007
Posts: 658
Location: Paris, France

> **eleven wrote:**
> okay, replace all exit by return and remove the last line.

It works. Thanks a lot.

Back to top   [profile] [pm] [www]

---

coloin   Posted: Mon Jul 06, 2009 5:36 am   Post subject: [quote]

Joined: 05 May 2005
Posts: 1007
Location: Oxford

The stats I gave were for the TOTAL minimal puzzles in the 40-subgrid.

It is an exercise to find a 31 - at least by random production.

There are only 2 obigatory clues, and the unavoidable sets can be seen in the complete solution grid - the 8@r3c9 and the 1@r7c1

Code:

```
+---+---+---+     +---+---+---+     +---+---+---+
|4..|3.7|.9.|     |465|387|192|     |...|.8.|..2|
|2.7|15.|6.3|     |287|159|643|     |...|...|...|
|...|..4|5.8|     |319|624|578|     |31.|.2.|..8|
+---+---+---+     +---+---+---+     +---+---+---+
|8.1|9..|7..|     |821|943|756|     |...|...|...|
|97.|2.5|.84|     |976|215|384|     |...|...|...|
|..4|..6|.19|     |534|876|219|     |53.|...|...|
+---+---+---+     +---+---+---+     +---+---+---+
|1.3|4..|..7|     |153|492|867|     |15.|...|...|
|.4.|.68|9..|     |742|568|931|     |...|...|...|
|6.8|.31|..5|     |698|731|425|     |...|...|...|
+---+---+---+     +---+---+---+     +---+---+---+
```

starting from the 40, and removing clues to minimality at 30 clues, the different nonminimal puzzles at each step is difficult for me to quantify, certainly there is a double count + at each step.
Edit.....its 1,10,45,120,210,252,210,120,45,10,1

To make a minimal 25 clue puzzle from a full grid requires the removal of 56 non-essential clues - in any order.
To make a minimal 24 clue puzzle from a full grid requires the removal of 57 non-essential clues - in any order.

So whats our fudge factor ?

C

Back to top   [profile] [pm]

---

ronk   Posted: Mon Jul 06, 2009 6:26 am   Post subject: [quote]

*** off-topic, I suppose ***

Joined: 02 Nov 2005
Posts: 2387
Location: Southeastern
USA

> **coloin wrote:**
> There are only 2 obigatory clues, and the unavoidable sets can be seen in the complete solution grid - the 8@r3c9 and the 1@r7c1

A grid with only two unavoidable sets is a surprise to me. Is it possible for a grid to have no unavoidable set? Just one?

Back to top   [profile] [pm]

---

coloin   Posted: Mon Jul 06, 2009 6:36 am   Post subject: [quote]

Joined: 05 May 2005
Posts: 1007
Location: Oxford

⚠️
These are the unavoidable sets left uncovered by the removal of the obligatory clues !!!!!!! [this shows why they are obligatory]
There are hundreds of thousands of unavoidable sets in every grid.

at the 35 clue stage there are only 252 non-minimal puzzles which have the 30-puzzle within
at the 35 clue stage there are 15505 non-minimal puzzles which have the 20-puzzle within
[out of a total of 263809 non-minimal 35-puzzles]

Thats why its easier to find the 20-clue puzzle in the 40-grid.

Extending this to 24/25 clues

A complete 81-clue grid
Remove 4 clues
The prob. that the subgrid still has got all the 24 clues from a puzzle is $57/81 * 56/80 * 55/79 * 54/78 \sim 57/k$
The prob. that the subgrid still has got all the 25 clues from a puzzle is $56/81 * 55/80 * 54/79 * 53/78 \sim 53/k$

This looks like the beginning of a fudge factor !

C

Back to top   👤 profile   ✉ pm

David P Bird   🗋 Posted: Mon Jul 06, 2009 8:38 am   Post subject: 💬 quote

Joined: 16 Sep 2008
Posts: 58
Location: Middle
England

FWIW here's a puzzle grid generation scheme I have manually tried several times now without reaching any invalid grids - mind you, I don't run at 2GHz and for random selection read "mental coin toss" so I can't really claim it's infallible.

1) Randomly fill boxes 1 & 5 with digits 1 to 9
2) Randomly select digits to occupy r7c7, r8c8 and r9c9.

From this point on, simple Sudoku eliminations must be made as each new digit is placed to reduce the cells sets available for further random placements.
The general aim is to take the digits one by one and position them randomly in the cells open to them.
The order in which the open cells are selected for the current digit is determined by prioritising the houses with most restricted choice first.

3) For the digit in r7c7 randomly select one of the available cells open for it in boxes 2,3,4, & 7 forcing its position in boxes 6 and 8
4) Repeat for digit in r8c8 in a box order decided by restricted choice
5) Repeat for digit in r9c9

6) Now same restricted choice principle is used to determine the order in which the remaining digits are placed, and where there is a choice, it is decided randomly.
Priority 1: Any digit which has already been forced as a single as a result of the previous placements
Priority 2: Any digit which is bi-local in any house.
Priority 3: One of the digits in any bivalue cell.
Priority 4: One of the digits in the first unresolved cell in box 2

Braid analysis logic lies at the back of this scheme which tells us that the same travelling pair can't exist both in a tier and a stack. The selection order used simply avoids such possibilities. Should you try it out, watch out for the emergence of unconditional sets towards the end.

I can't see that this system is biased, but then I know I am, so I wait for the opinions of others.

Back to top   👤 profile   ✉ pm

eleven   🗋 Posted: Mon Jul 06, 2009 9:33 am   Post subject: 💬 quote

Joined: 10 Feb 2008
Posts: 319

coloin wrote:
> Almost certainly they are both biased because the "real" mean puzzle size is >25 clues.

Maybe its worth to try this: Take bunches of random 23, 24, 25 and 26 clues puzzles and do a {-1,+2} and {-2,+1} on them. If you get more (n+1) puzzles than (n - 1) puzzles, i guess its probable, that the mean value is higher (?).
Coloin, whats the command line for gsf's program to do that ?

Back to top   👤 profile   ✉ pm

Red Ed   🗋 Posted: Mon Jul 06, 2009 9:34 am   Post subject: 💬 quote

Joined: 06 Jun 2005
Posts: 540

For Denis:

Code:

```
static char *G ="\n"
"\n"
"    suexg version 14, small randomized sudoku-generator in C,\n"
"\n"
"    generates about 24 sudokus per second with 1GHz\n"
"    based on an exact cover solver, compiled with gcc3.2\n"
"    Report bugs,improvement suggestions,feedback to sterten@aol.com\n"
```

```
"    For some explanation of the solver see: http://magictour.free.fr/suexco.txt\n"
"    This generator starts from an empty grid and adds clues completely at random\n"
"    There are faster pseudo-random methods which generate upto 1000 sudokus\n"
"    per second.\n"
"    For a solver see: http://magictour.free.fr/suexk.exe\n"
"      (C-sourcecode is attached to the executable )\n"
"    Send sudokus with rating more than 100000 to sterten@aol.com\n"
"    so they can be included in the list of hardest sudokus at\n"
"    http://magictour.free.fr/top94\n"
"\n"
"    You can download a DOS/Windows executable of this program from\n"
"    http://magictour.free.fr/suexg.exe\n"
"\n"
"\n"
"    This software is public domain\n"
"\n";


#include <stdio.h>
#include <stdlib.h>

#define MWC ((zr=36969*(zr&65535)+(zr>>16))^(wr=18000*(wr&65535)+(wr>>16)))

unsigned zr=362436069, wr=521288629;
int Rows[325],Cols[730],Row[325][10],Col[730][5],Ur[730],Uc[325],V[325],W[325];
int P[88],A[88],C[88],I[88],Two[888];
// char B[83]="011115555513555999913599988813394777811394667833344267834442267844222667822266778";
char B[83]="011122233311112223331112223334445556666444555666644455566677788899977788899977788899";
char H[326][7];
int b,w,f,s1,m0,c1,c2,r1,l,i1,m1,m2,a,p,i,j,k,r,c,d,n=729,m=324,x,y,s;
int mi1,mi2,q7,part,nt,rate,nodes,seed,solutions,min,samples,sam1,clues;
char L[99]=".123456789";
FILE *file;

int solve();

//------------------------------------------------------------
int main(int argc,char*argv[]){
  if(argc<2){printf("\nusage:suexg random-seed [z] [with rating] \n\n");
        printf("      generates z locally minimal sudokus [and rates them]\n");
        printf("      use different numbers for seed to get different streams of sudokus\n");
        printf("      default is : z=1e9 and without rating   [if these are not specified]\n");
        printf("      to redirect the sudokus to a file use e.g. : suexg 0 100 r >file\n");
        printf("\n   suexg i : printf more info\n     suexg s  : prints source-code\n");
        exit(1);;}
  sscanf(argv[1],"%i",&seed);zr^=seed;wr+=seed;
  samples=1000000000;if(argc>2)sscanf(argv[2],"%i",&samples);
  rate=0;if(argc>3)rate=1;if(argc>4)rate=2;
  if(argv[1][0]=='i'){printf("%s",G);exit(0);}
  if(argv[1][0]=='s'){if((file=fopen(argv[0],"rb"))==NULL)
                        {printf("\ncan't find suexg.exe\n");exit(1);}
     ip1:w=0;for(i=1;i<33;i++)w+=(fgetc(file)==45);if(w<32)goto ip1;
     while(fgetc(file)!='_');
     ip2:x=fgetc(file);if(x!=13)printf("%c",x);if(feof(file))exit(1);goto ip2;}

for(i=0;i<888;i++){j=1;while(j<=i)j+=j;Two[i]=j-1;}

r=0;for(x=1;x<=9;x++)for(y=1;y<=9;y++)for(s=1;s<=9;s++){
r++;Cols[r]=4;Col[r][1]=x*9-9+y;Col[r][2]=(B[x*9-9+y]-48)*9-9+s+81;
Col[r][3]=x*9-9+s+81*2;Col[r][4]=y*9-9+s+81*3;}
for(c=1;c<=m;c++)Rows[c]=0;
for(r=1;r<=n;r++)for(c=1;c<=Cols[r];c++){
a=Col[r][c];Rows[a]++;Row[a][Rows[a]]=r;}

c=0;for(x=1;x<=9;x++)for(y=1;y<=9;y++){c++;H[c][0]='r';H[c][1]=x+48;H[c][2]='c';H[c][3]=y+48;H[c][4]=0;}
c=81;for(b=1;b<=9;b++)for(s=1;s<=9;s++){c++;H[c][0]='b';H[c][1]=b+48;H[c][2]='s';H[c][3]=s+48;H[c][4]=0;}
c=81*2;for(x=1;x<=9;x++)for(s=1;s<=9;s++){c++;H[c][0]='r';H[c][1]=x+48;H[c][2]='s';H[c][3]=s+48;H[c][4]=0;}
c=81*3;for(y=1;y<=9;y++)for(s=1;s<=9;s++){c++;H[c][0]='c';H[c][1]=y+48;H[c][2]='s';H[c][3]=s+48;H[c][4]=0;}


    sam1=0;
m0s:sam1++;if(sam1>samples)exit(0);

m0: for(i=1;i<=81;i++)A[i]=0;part=0;q7=0;
mr1:i1=(MWC>>8)&127;if(i1>80)goto mr1;i1++;if(A[i1])goto mr1;
mr3:s=(MWC>>9)&15;if(s>8)goto mr3;s++;
    A[i1]=s;m2=solve();q7++;//if(q7>999)goto m0;
// add a random clue and solve it. No solution ==> remove it again.
// Not yet a unique solution ==> continue adding clues
    if(m2<1)A[i1]=0;if(m2!=1)goto mr1;

//now we have a unique-solution sudoku. Now remove clues to make it minimal
part++;if(solve()!=1)goto m0;
for(i=1;i<=81;i++){mr4:x=(MWC>>8)&127;if(x>=i)goto mr4;x++;P[i]=P[x];P[x]=i;}
for(i1=1;i1<=81;i1++){s1=A[P[i1]];A[P[i1]]=0;if(solve()>1)A[P[i1]]=s1;}

if(rate){nt=0;mi1=9999;for(f=0;f<100;f++){solve();nt+=nodes;if(nodes<mi1)
{mi1=nodes;mi2=C[clues];}}
printf("rating:%6i ,  ",nt);if(rate>1)printf("hint:%s    ",H[mi2]);}
for(i=1;i<=81;i++)printf("%c",L[A[i]]);printf("\n");
goto m0s;}


//------------------------------------------------------------------------
int solve(){//returns 0 (no solution), 1 (unique sol.), 2 (more than one sol.)
   for(i=0;i<=n;i++)Ur[i]=0;for(i=0;i<=m;i++)Uc[i]=0;
   clues=0;for(i=1;i<=81;i++)
     if(A[i]){clues++;r=i*9-9+A[i];
        for(j=1;j<=Cols[r];j++){d=Col[r][j];if(Uc[d])return 0;Uc[d]++;
```

```
        for(k=1;k<=Rows[d];k++){Ur[Row[d][k]]++;}}}
    for(c=1;c<=m;c++){V[c]=0;for(r=1;r<=Rows[c];r++)if(Ur[Row[c][r]]==0)V[c]++;}

    i=clues;m0=0;m1=0;solutions=0;nodes=0;
m2:i++;I[i]=0;min=n+1;if(i>81 || m0)goto m4;
    if(m1){C[i]=m1;goto m3;}
    w=0;for(c=1;c<=m;c++)if(!Uc[c]) {  if(V[c]<2){C[i]=c;goto m3;}
            if(V[c]<=min){w++;W[w]=c;};
            if(V[c]<min){w=1;W[w]=c;min=V[c];} }
mr:c2=MWC&Two[w];if(c2>=w)goto mr;C[i]=W[c2+1];
m3:c=C[i];I[i]++;if(I[i]>Rows[c])goto m4;
    r=Row[c][I[i]];if(Ur[r])goto m3;m0=0;m1=0;
    nodes++;//if(nodes>9999 && part==0)return 0;
    for(j=1;j<=Cols[r];j++){c1=Col[r][j];Uc[c1]++;}
    for(j=1;j<=Cols[r];j++){c1=Col[r][j];
        for(k=1;k<=Rows[c1];k++){r1=Row[c1][k];Ur[r1]++;if(Ur[r1]==1)
            for(l=1;l<=Cols[r1];l++){c2=Col[r1][l];V[c2]--;
                if(Uc[c2]+V[c2]<1)m0=c2;if(Uc[c2]==0 && V[c2]<2)m1=c2;}}}
    if(i==81)solutions++;if(solutions>1)goto m9;goto m2;
m4:i--;c=C[i];r=Row[c][I[i]];if(i==clues)goto m9;
    for(j=1;j<=Cols[r];j++){c1=Col[r][j];Uc[c1]--;
        for(k=1;k<=Rows[c1];k++){r1=Row[c1][k];Ur[r1]--;
            if(Ur[r1]==0)for(l=1;l<=Cols[r1];l++){c2=Col[r1][l];V[c2]++;}}}
    if(i>clues)goto m3;
m9:return solutions;}
```

Back to top [profile] [pm]

---

coloin

Joined: 05 May 2005
Posts: 1007
Location: Oxford

Posted: Mon Jul 06, 2009 11:52 am   Post subject: [quote]

Red Ed......thats a good idea.

sudoku -go{-1+2}{-2+1} would do it, although non-minimals will also be produced at both stages.

Maybe the average number of puzzles produced by a {-2+2} for 24,25,26-puzzles would also do the same thing.

Havards program can excludes minimals.

I will try the {-2+2}

Back to top [profile] [pm]

Red Ed

Joined: 06 Jun 2005
Posts: 540

Posted: Mon Jul 06, 2009 12:11 pm   Post subject: [quote]

coloin wrote:
> **Red Ed**......thats a good idea.

I think you mean **eleven** ...

Back to top [profile] [pm]

---

ronk

Joined: 02 Nov 2005
Posts: 2387
Location: Southeastern USA

Posted: Mon Jul 06, 2009 3:46 pm   Post subject: [quote]

denis_berthier wrote:
> E.g. r <-> n super-symmetry changes LS to HS.
> ...
> There was something imprecise in my post.
> I corrected it.
> ...
> Notice that, for complementarity and symmetry reasons, an ALS(5) or an AHS(5) are counted as a LS(4), an ALS(6) or an AHS(6) as a LS(3) ...
> ...
> Does this answer your question?

It appears your "complementarity" differs from the traditional complementary LS-HS relationship, so it really doesn't matter.

Back to top [profile] [pm]

Display posts from previous: [All Posts] [Oldest First] [Go]

[newtopic] [postreply]   **Sudoku Players' Forums Forum Index -> Advanced solving techniques**

All times are GMT - 8 Hours

Jump to: [Advanced solving techniques] [Go]

Powered by phpBB © 2001, 2005 phpBB Group