

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head><title>Error-Detection Magic Illuminated for Frontiers In Education conference special session, Oct '18</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" name = "keywords" />
<style> h1 {font-size: inherit;} body {background-color: #BFBF00}
  table {float: left; border-spacing: 1px} th {background-color: #BFBF00; padding: 0} td {text-align: center}
  .b1 {background-color: white; color: black} .b0 {background-color: black; color: white} .h0 {visibility: hidden}
  #illumelt {font-size: 70%}</style>
<script type="text/javascript">
function gid(id) { return(document.getElementById(id)); }
function gup(name) { /* Get URL query string parameter */ name = name.replace(/[\/\?&#]/, "\\[/\?&#]");
  var regexS = "[\\?&]" + name + "=[^&#]*"; var regex = new RegExp( regexS );
  var results = regex.exec( window.location.href ); if( results == null ) return ""; else return results[1]; }
v=gup('v'); if (!v) v='1'; cheating=0; document.getElementsByTagName('title')[0].innerHTML += ' : Trick Version '+v;
illum = Number(gup('illum')); /* non-zero for true; 0 or non-number for false */
switch (v) { /* Branching on Version number of trick; round n up to 1 mod 3 for version 2b */
case '1': n=Number(gup('n')); if (!n) n=6; if (n<3) n=3; maxflips=0; break;
case '2a': n=Number(gup('n')); if (!n) n=8; if (n<4) n=4; maxflips=Math.floor((n-1)/2); break;
case '2b': n=Number(gup('n')); if (!n) n=10; if (n<4) n=4; n=n+2-(n+1)%3; maxflips=(n-1)/3; break;
case '3': n=Number(gup('n')); n=10; cheating=1; maxflips=(n-1)/3; break;
case '4': n=Number(gup('n')); n=11; cheating=1; maxflips=1; break; /* Maybe allow other n later */
case '5': d=Number(gup('d')); if (!d) d=3; if (d<2) d=2; n=1; for (i=1;i<=d;++i) n=n*2; maxflips=1; } /* n=2^d */
numcells=n*n; if (v==5) dimsiz=4; else { d=2; dimsiz=n; }; linespergroup=((v==1 || v=='2a')? dimsiz-1: 3);
htmlwidth = document.getElementsByTagName('html')[0].offsetWidth;
twidth = Math.floor(.67*htmlwidth/n); theight = Math.max(7,twidth/2); fontsize = .3*htmlwidth/165;
var sizes = {p:fontsize, input:fontsize/3, td:Math.max(1,twidth/64), th:Math.max(0,twidth/64)};
for (elt in sizes) document.write('<style> '+elt+' {font-size: '+sizes[elt]+'em}</style>');
function D1toD2(num) {return('Row'+Math.floor(num/n)+' ,Column'+num%n);} function D2toD1(row, col) {return(row*n+col);}
function bwgrid() { groupingp=(d==2)&&(v!=4); document.write('<table><tr><th></th>');
  for (col=0;col<n;++col) document.write('<th id="col'+col+'">'+col+'</th>'); document.write('</tr>'); cell=0;
  for (row=0;row<n;++row) { document.write('<tr><th id="row'+row+'">'+row+'</th>');
    for (col=0;col<n;++col) document.write('<td id="'+cell++'" width="'+twidth+'px" height="'+theight+'px" class=
"b'+Math.floor(2*Math.random())+' ' + (groupingp?'style="border-left: '+col%linespergroup==0?'solid #BFBF00 5px':
'none')+'; border-top: +(row%linespergroup==0?'solid #BFBF00 5px':'none')+'; " : "' + 'onClick="Flip(this)"></td>');
    document.write('</tr>'); }
  document.write('</table>');
  if (v=='1') { gid('col'+(n-1)).className = 'h0'; gid('row'+(n-1)).className = 'h0'; /* header ends */
    for (rc=0;rc<n;++rc) gid(D2toD1(rc,n-1)).className = (gid(D2toD1(n-1,rc)).className = 'h0'); /*ends*/ }
function value(cell) { return(parseInt(gid(cell).className.charAt(1))); }
var saveillum=''; var orderstr = ''; function ordermsg(msg) { if (orderstr=='') orderstr=msg; }
var flipcells = new Array(maxflips); for (i=0;i<maxflips;++i) flipcells[i]=0;
quotient35=0; mod35=0; /* For v==3, need these and above array to be global to flip and organize */
function doparities() { var parities = new Array(dimsiz); flipinds = new Array(dimsiz/* use < 1/2 or 1/3 */);
  divisor=1; for (dim=d-1;dim>0;--dim) { for (i=0;i<dimsiz;++i) parities[i]=0; for (cell=0;cell<numcells;++cell) {
  index = Math.floor(cell/divisor)%dimsiz; parities[index] = parities[index] ^ value(cell); }
  if (phase=='organizing' && v=='1') {
  if (dim==0) { for (row=0;row<n-1;++row) {gid(D2toD1(row,n-1)).className='b'+parities[row]; parities[row]=0;
  gid('row'+(n-1)).className = ''; }
  else { for (col=0;col<n;++col) {gid(D2toD1(n-1,col)).className='b'+parities[col]; parities[col]=0;
  gid('col'+(n-1)).className = ''; } }
  parsum=0; parmaj=0; flipictr=0; flipinds[0]=dimsiz-1;
  for (line=0;line<dimsiz-1;++line) { parsum += parities[line];
  if ((line+1)%linespergroup==0) { parmaj=0; if (parsum>linespergroup/2) parmaj=1; parsum=0;
  for (gline=line+1-linespergroup; gline<=line; ++gline) {
  if (parities[gline]!=parmaj) flipinds[flipictr++]=gline; }}}
  if (phase=='organizing' && v=='3') {
  place=((dim==1)?oneplace:sixplace); /* oneplace/sixplace will be encoded in order of flip cols/rows resp. */
  saveillum+='<br />For '+ (dim==1?'cols':'rows')+' , order of flip indices governed by '+place; orderstr='';
  switch(flipictr) {
  case 2: flipinds[2]=n-1; saveillum+=' (and add '+ (n-1)+' to two true flip indices)'; /*Fall through*/
  case 3: switch(place) { /*Code 0--5 by row/col order: 0=0,1,2; 1=0,2,1; 2=1,0,2; 3=1,2,0; 4=2,0,1; 5=2,1,0*/
  case 1: ordermsg('ordering indices as smallest, largest, middle (');
  temp=flipinds[1]; flipinds[1]=flipinds[2]; flipinds[2]=temp; break;
  case 3: ordermsg('ordering indices as middle, largest, smallest (');
  temp=flipinds[0]; flipinds[0]=flipinds[2]; flipinds[2]=temp; /* Fall through */
  case 2: ordermsg('ordering indices as middle, smallest, largest (');
  temp=flipinds[0]; flipinds[0]=flipinds[1]; flipinds[1]=temp; break;
  case 4: ordermsg('ordering indices as largest, smallest, middle (');
  temp=flipinds[0]; flipinds[0]=flipinds[1]; flipinds[1]=temp; /* Fall through */
  case 5: ordermsg('ordering indices as largest, middle, smallest (');
  temp=flipinds[0]; flipinds[0]=flipinds[2]; flipinds[2]=temp; break;}
  if (orderstr=='') orderstr='leaving indices in order smallest, middle, largest (';
  saveillum+='( '; encode by '+orderstr); break;
  case 0: flipinds[0]=n-1; saveillum+=' (and add '+ (n-1)+' to empty set of indices) '; /*Fall through*/
  case 1: /* When flipictr<=1, encode place by doubling a col */
  if (flipinds[0]<=place) {++place; saveillum+='Increment '+place+' due to true index in use. '};

```

```

flipinds[2]=(flipinds[1]=place); saveillum+='; encode by using twice as flip index (';
if (dim==1) { flipinds[2]=flipinds[0]; flipinds[0]=place; } /*changeup so do not flip same cell twice*/
saveillum+=((dim==1?'cols ':'rows')+flipinds[0]+' '+flipinds[1]+' '+flipinds[2]+' '); flipictr=3; }
if (illum) { for (cell=0;cell<numcells;++cell) { index = Math.floor(cell/divisor)%dimsiz;
celt=gid(cell); bordertopstyle=borderbottomstyle=bordersidestyle='hidden';
for (i=Math.max(0,flipictr-1);i>=0;--i) if (flipinds[i]==index) bordersidestyle='solid';
if (index<dimsiz-1) {if (parities[index]) {parcolor='magenta'; bordertopstyle='solid'
else {parcolor='blue'; borderbottomstyle='solid'}}
else parcolor='#BFBF00';
celt.innerHTML = '<span style="border-width: 1px; border-color: #BFBF00; border-style: ' + bordertopstyle
+ ' ' +bordersidestyle + ' ' +borderbottomstyle + ' '; color: ' + parcolor + '">' + index + '</span>'
+ ( (celt.innerHTML=='') ? '' : '<span style="color: #BFBF00;"></span>' ) + celt.innerHTML; }
for (i=maxflips-1;i>=0;--i) { flipcells[i]+=(i<flipictr)?flipinds[i]:(dimsiz-1)*divisor; }
divisor = divisor*dimsiz; }
illumelt.innerHTML = (phase=='organizing'?saveillum:'' )
+ ( phase=='organizing' ||phase=='testingLIFO' ||phase=='testingmain' ? '<br />Cells show indices in each
of the '+d+ ' dimensions: even parity indicated by underlined <span style="color: blue">blue</span> & odd parity by
overlined <span style="color: magenta">magenta</span>. Outlier/extra indices are between vertical lines.' : '' ); }
function clearclues() { illumelt.innerHTML=''; tds=document.getElementsByTagName('td');
for (td in tds) if(tds[td].innerHTML!='X') tds[td].innerHTML = '';}
function Flip(tdelt) { /* Flips a tile as long as not locked out */
switch (phase) { /* Branch according to what stage of the trick we are at */
case 'testingLIFO':
if (tdelt==LIFOflip) {tdelt.innerHTML='YES.'; direlt.innerHTML = 'The wizard will reveal the other flip.';
phase='testingmain'; clearclues(); illumelt.innerHTML='Recall: '+saveillum;
if (quotient35>0) {++drumnum; if (quotient35>1) ++drumnum; drumrolls(); }}
else tdelt.innerHTML='NO.'; return;
case 'testingmain': if (tdelt==mainflip) tdelt.innerHTML='YES'; else tdelt.innerHTML='NO'; return;
case 'lockout': return;
case 'mainflip': mainflip = tdelt; clearclues(); tdelt.innerHTML = 'Remember '+D1toD2(tdelt.id); phase='lockout';
if (cheating) direlt.innerHTML = 'Click the button below to generate organizing flips:<br/><input type="button"
value="Organize" onClick="Organize()">'; else direlt.innerHTML = 'Click the button below to submit to wizard (who
will tell the tile flipped):<br /><input type="button" value="Submit" onClick="Testwizard()">'; break;
case 'LIFOflip': LIFOflip = tdelt; clearclues(); tdelt.innerHTML = 'Remember also '+D1toD2(tdelt.id);
phase='lockout'; direlt.innerHTML='Click the button below to submit to wizard (who will tell the tile flipped):
<br /><input type="button" value="Submit" onClick="Testwizard()">'; break;
case 'organizing':
if (tdelt.innerHTML!='X') return; clearclues(); tdelt.innerHTML=''; --num2organize;
if (num2organize==0) {
if (v!=='4') { phase='mainflip'; if (v=='3') phase='LIFOflip';
direlt.innerHTML = 'Now click <em>one</em> tile of your own to flip.'; }
else { direlt.innerHTML = 'Now the wizard will reveal your earlier flip.'; phase='testingmain';
orgrow=9*mfrow%11; orgcol=8*mfcol%11; illumelt.innerHTML = 'Since the organizing flip was (' + orgrow +
',' + orgcol + '), we reveal (5*' + orgrow + ' mod 11,7*' + orgcol + ' mod 11)'; }}
else if (v=='3') { illumelt.innerHTML='Recall: '+saveillum; gid(flipcells[3-num2organize]).innerHTML="X"; }}
tdelt.className = 'b'+(1-tdelt.className.charAt(1)); }
saveddirs=''; drumnum=0; function drumrolls() { saveddirs=direlt.innerHTML; direlt.innerHTML+='<p>Drumroll please!';
--drumnum; setTimeout(endroll,6000); } /* Variables global to drumroll and endroll functions */
function endroll() {direlt.innerHTML = saveddirs; if (drumnum>0) setTimeout(drumrolls,5000);}
function Testwizard() { clearclues(); phase='testingmain'; direlt.innerHTML='The wizard will tell the tile flipped.';
if (v=='3') { phase='testingLIFO'; drumnum=1; setTimeout(drumrolls,500); } doparities(); }
function Organize() { phase='organizing'; switch (v){ /* Branching on Version number of trick */
case '4': gid(mainflip.id).innerHTML=''; mfrow=Math.floor(mainflip.id/n); mfcol=mainflip.id%n; num2organize=1;
gid(D2toD1((9*mfrow%11,(8*mfcol%11)).innerHTML="X"; /* 9 is 5^{ -1} mod 11; 8 is 7^{ -1} mod 11 */
direlt.innerHTML = 'Please announce and perform the indicated flip.'; illumelt.innerHTML =
'With secret flip ('+mfrow+', '+mfcol+', computer marks (9*mfrow+ mod 11,8*mfcol+ mod 11)'; break;
case '3': mainflip.innerHTML=''; mod35=mainflip.id%35; /* Code in base 6 to order organizing flips: */
sixplace=Math.floor(mod35/6); oneplace=mod35%6; quotient35/*will be #drumrolls*/=Math.floor(mainflip.id/35);
saveillum = 'First flip ' + mainflip.id + '=' + mod35+ '+' + '35*' + quotient35 + ' (# of ending drumrolls), and
' + mod35 + ' base 10 = ' + sixplace + oneplace + ' base 6; use to arrange flip rows & cols:' /* Falls through */
default: doparities(); if (v=='3') { num2organize=3; gid(flipcells[0]).innerHTML = "X";
direlt.innerHTML = 'Please announce and perform indicated flip.'; }
else { for (num2organize=0;num2organize<maxflips;++num2organize) {
if (flipcells[num2organize]<numcells-1) gid(flipcells[num2organize]).innerHTML='X'; else break; }
if (num2organize>0) direlt.innerHTML = "The tile" + (num2organize>1?"s":"") + " with an X offend the
wizard\'s sense of organization;<br /> please click " + (num2organize>1?"them":"it") + " to flip.";
else { phase='mainflip'; direlt.innerHTML='The wizard '+ (v=='1') ? 'added tiles to make the array bigger'
: 'is happy with this organization' ) + '. While the wizard is not looking, click <em>one</em> tile to flip.'; }}}
function Ready() { direlt.innerHTML='Click a tile the wizard will reveal later:<br />'; phase='mainflip'; }
</script>
</head><body><h1>Version </h1><script type="text/javascript"> bwgrid(); </script><p><span id="direlt"></span><br />
<span id="illumelt"></span></p> <script type="text/javascript"> document.getElementsByTagName('h1')[0].innerHTML+=v;
id= gid('direlt'); illumelt = gid('illumelt'); phase='initial'; bval= (cheating ? 'Ready' : 'Organize');
illumelt.style.visibility=(illum?"visible":"hidden");
direlt.innerHTML = 'Click any tiles you would like to flip,<br /> or reload to randomize.<br />Then click the
button below:<br /><input type="button" value="" + bval + " onClick="" + bval + '()'>; </script> </body> </html>

```