**Pseudocode**

start subprocedure randomiseBoard() //no input or output  
// input board array represented as 3 x 3 grid  
// BSZ=3 BSZ2=BSZ\*BSZ  
:initialise shuffleArr with length BSZ2 and set all elements to zero  
:call shuffleArray(shuffleArr, BSZ2);  
:declare board[BSZ2];  
for(int i=0; i<BSZ2; i++) begin  
 :board[i]=shuffleArr[i];  
end for  
end function

start procedure getChoice() returns number between 1 and 8  
:cout << Enter number between 1 and 8;  
:declare choice as int;  
:cin >> choice;  
:return choice;  
end function

begin function -> shuffle\_array  
 :input -> output\_array, arraylen;  
 :declare -> arraylen as length of ouput\_array;  
 :set output\_array values to zero;  
 :declare -> i=0;  
 :declare -> j=random\_between\_1\_and\_arraylen;  
 while(i<arraylen) is yes then  
 while(output\_array[j]>0) is yes then  
 :j=random\_between\_1\_and\_arraylen;  
 endwhile is no  
 :output\_array[j]=i;  
 :i=i+1;  
 endwhile is no  
endfunction

procedure checkChoice()

start

: let choice be int v;  
if (v is not between 0 to 8) then (yes)  
: return false;  
else(no)  
endif  
: let index of 0 be id0 and index of v be idv;  
if (idv - id0 is not equal to (-3 or 3 or -1 or 1)) then (yes)  
: return false;  
else(no)  
endif  
: return true;

stop

procedure drawBoard()

start

:let BSZ=3;  
:i =0;  
while (i less then BSZ) is (yes)  
:j=0;

while (j less then BSZ) is (yes)

:display board [ i multiplied by BSZ + j];  
:j++;  
end while (no)

:display new line;  
:i++;  
end while (no)

stop

procedure checkWin()  
Start

if(not [first space and last space has 0 or 1 for first and0 or 8 for last ]) then (yes)  
:winner=false;  
note right: step 1 check first and last board;  
: return;  
else(no)  
endif  
:int i=1;  
note right: step 2 check remaining board spaces

while(i<BSZ2-1) is (yes)

if(not (board[i]>board[i-1])) then (yes)

:winner=false;

:return;  
else(no)  
endif  
:i++;

end while (no)

:winner=true;  
:return;  
Stop