ALGORITHM

STEP 1: [Begin] Start

STEP 2: Display “SUDOKU INSTRUCTIONS”

STEP 3: [Make Sudoku Matrix Empty]

for i=0 to 8 in steps of 1

do

for j=0 to 8 in steps of 1

do

S[i][j] 🡨 NULL

Done

Done

STEP 4: Initialize fix values

STEP 5: [Define fix values]

for i=0 to 8 in steps of 1

do

for j=0 to 8 in steps of 1

do

if(S[i][j]==NULL) then fix[i][j] 🡨 0

otherwise fix[i][j] 🡨1

done

done

STEP 6: [Initialize p,q]

p 🡨 4

q 🡨 4

STEP 7: [Display Sudoku Matrix]

ascii 🡨 (int)ch

for i = 0 to 8 in steps of 1

do

print ("\n|")

for j = 0 to 8 in steps of 1

do

if(i==p && j==q && fix[i][j]==0 && ascii>=49 && ascii<=57)

Print (” S[i][j]”)

else if(i==p&&j==q && fix[i][j]==0)

Print (“\_”)

else if(S[i][j]==NULL)

Print (“ ”)

otherwise

Print (“ S[i][j]”)

if(j==2||j==5)

Print (" ||")

otherwise

Print (" |")

done

Print (“\n”)

done

Print " ( p+1,q+1) "

STEP 7: [Check whether Matrix is complete]

for i=0 to 8 in steps of 1

do

for j=0 to 8 in steps of 1

do

if(S[i][j]==NULL)

go to STEP 9

done

done

STEP 8: [Check whether Sudoku is Correct]

For i=0 to 8 in steps of 1

do

empt\_flag()

for j=0 to 8 in steps if 1

do

k=S[i][j]

flag[k-1]=1

done

row[i]=check1(flag,9)

if(row[i]==0)

go to STEP 9

done

For i=0 to 8 in steps of 1

do

empt\_flag()

for j=0 to 8 in steps if 1

do

k=S[j][i]

flag[k-1]=1

done

column[i]=check1(flag,9)

if(column[i]==0)

go to STEP 9

done

check\_box(0,2,0,2)

box[0] 🡨 check1(flag,9)

check\_box(0,2,3,5)

box[1] 🡨 check1(flag,9)

check\_box(0,2,6,8)

box[2] 🡨 check1(flag,9)

check\_box(3,5,0,2)

box[3] 🡨 check1(flag,9)

check\_box(3,5,3,5)

box[4] 🡨 check1(flag,9)

check\_box(3,5,6,8)

box[5] 🡨 check1(flag,9)

check\_box(6,8,0,2)

box[6] 🡨 check1(flag,9)

check\_box(6,8,3,5)

box[7] 🡨 check1(flag,9)

check\_box(6,8,6,8)

box[8] 🡨 check1(flag,9)

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

do

if(box[i]==0)

go to STEP 9

done

Print “You Won”

goto step 15

STEP 9: Read ch

STEP 10: [Check ch]

switch(ch)

do

case ‘a’ : [Moves position of curser left]

if(q==0)

goto step 9

else

do

count 🡨 0

for i=q-1 till fix[p][i]==1 in steps of 1(decrement)

count++

if(i==-1)

goto step 9

else

q 🡨 q-count-1

done

break

case 'd': [Moves position of curser right]

if(q==8)

goto step 9

else

do

count 🡨 0

for i=q+1 till fix[p][i]==1 in steps of 1(increment)

count++

if(i==9)

goto step 9

else

q=q+count+1

done

break

case 'w': [Moves position of curser upward]

if(p==0)

goto ask;

else

do

Count 🡨 0;

for i=p-1 till fix[i][q]==1 in steps of 1(decrement)

count++

if(i==-1)

goto step 9

else

p=p-count-1

done

break

case 's': [Moves position of curser downward]

if(p==8)

goto step 9

else

do

count 🡨 0

for i=p+1 till fix[i][q]==1 in steps of 1(increment)

count++

if(i==9)

goto step 9

else

p=p+count+1

done

break

case '0': [Vanishes the previosly assigned value at curser position]

if(S[p][q]!=NULL)

S[p][q] 🡨 NULL

else

goto step 9

break

case 1 to 9: S[i][j] 🡨 1 to 9 respectively

case '.': goto step 12

otherwise goto step 9

done

STEP 11: goto step 7

STEP 12: [Displays "OPTIONS"]

Print(" >> OPTIONS <<

1. Press c or C to close

Once closed, your previously filled numbers will disappear.

2. Press b or B to go back

3. Press r or R to restart the game “)

STEP 13: [Input Option]

Read decide

STEP 14: [Check option]

if(decide=='c' or decide=='C')

goto step 15

else if(decide=='b' or decide=='B')

goto step 7

else if(decide=='R' or decide=='r')

goto step 3

else

do

Print"\n Invalid Answer..."

Print"\n Give your answer: "

goto step 13

done

STEP 14: Stop