#!/bin/sh

Width=50

Height=15

PadX=10

PadY=5

Speed="2"

## 蛇起始位置

((SnakeX=PadX+Width/2))

((SnakeY=PadY+Height/2))

((ScoreX=PadX+Width/2-10))

((ScoreY=PadY-3))

score=0

SnakeHead="\033[41m@\033[0m"

SnakeBody="\033[41m \033[0m"

P1="\033[31mO\033[0m"

P2="\033[32mO\033[0m"

P3="\033[33mO\033[0m"

P4="\033[34mO\033[0m"

P5="\033[35mO\033[0m"

P6="\033[36mO\033[0m"

P7="\033[37mO\033[0m"

PS=(" " $P1 $P2 $P3 $P4 $P5 $P6 $P7)

PNum=${#PS[\*]}

clear

function DrawBox(){

local XSmall XBig YSmall YBig

((XSmall=PadX-2))

((YSmall=PadY-1))

((XBig=PadX+Width))

((YBig=PadY+Height))

for((i=XSmall;i<=XBig;i+=2))

do

echo -ne "\033[$YSmall;${i}H\033[42m[]\033[0m"

echo -ne "\033[$YBig;${i}H\033[42m[]\033[0m"

done

for((i=YSmall;i<=YBig;i++))

do

echo -ne "\033[$i;${XSmall}H\033[42m[]\033[0m"

echo -ne "\033[$i;${XBig}H\033[42m[]\033[0m"

done

echo

echo

}

function CordToKey(){

local x y Max

Max=100

x=$1

y=$2

((x+=Max))

((y+=Max))

echo $x$y

}

function Values(){

local i j

for((i=PadX;i<=PadX+Width;i++))

do

for((j=PadY;j<=PadY+Height;j++))

do

values[`CordToKey $i $j`]="$i|$j"

done

done

}

function GameOver(){

local x y

((x=PadX+Width/2-5))

((y=PadY+Height+2))

echo -e "\033[$y;${x}H \033[32mGame Over!\033[0m\n\n"

kill $PPID

MoveXYExit

}

function NewP(){

local x y p v

while :

do

((x=RANDOM%Width+PadX))

((y=RANDOM%Height+PadY))

v=${values[`CordToKey $x $y`]}

if [[ $v =~ "\|" ]];then

((p=RANDOM%((PNum-1))+1))

echo -ne "\033[$y;${x}H${PS[$p]}"

values[`CordToKey $x $y`]="$p"

break

fi

done

}

function Moving(){

local X Y oldX oldY v i j sx sy

X=$1; Y=$2; oldX=$3; oldY=$4; v=$5

echo -ne "\033[$Y;${X}H$SnakeHead"

values[`CordToKey $X $Y`]="snake"

Snake $X $Y

echo -ne "\033[$oldY;${oldX}H$SnakeBody"

if [[ ${#v} != 1 ]];then

for((i=0;i<${#SnakeValue[\*]};i+=2))

do

if [ "${SnakeValue[$i]}" != "" ];then

((j=i+1))

sx=${SnakeValue[$i]}

sy=${SnakeValue[$j]}

SnakeValue[$i]=""

SnakeValue[$j]=""

echo -ne "\033[$sy;${sx}H "

values[`CordToKey $sx $sy`]="$sx|$sy"

break

fi

done

else

((score+=v))

echo -ne "\033[$ScoreY;${ScoreX}H \033[32m Score: $score \033[0m"

NewP

fi

}

function Snake(){

SnakeValue[${#SnakeValue[\*]}]=$1

SnakeValue[${#SnakeValue[\*]}]=$2

}

function MoveXY(){

local sig oldX oldY v

Init

trap "sig=26" 26

trap "sig=27" 27

trap "sig=28" 28

trap "sig=29" 29

trap "MoveXYExit" 30

while :

do

oldX=$X

oldY=$Y

case $sig in

28)((maxX=Width+PadX-1))

((X++))

v=${values[`CordToKey $X $Y`]}

[[ $X -gt $maxX || "$v" == "snake" ]] && GameOver

;;

29)((X--))

v=${values[`CordToKey $X $Y`]}

[[ $X -lt $PadX || "$v" == "snake" ]] && GameOver

;;

27)((maxY=Height+PadY-1))

((Y++))

v=${values[`CordToKey $X $Y`]}

[[ $Y -gt $maxY || "$v" == "snake" ]] && GameOver

;;

26)((Y--))

v=${values[`CordToKey $X $Y`]}

[[ $Y -lt $PadY || "$v" == "snake" ]] && GameOver

;;

esac

Moving $X $Y $oldX $oldY $v

sleep .$Speed

done

}

function Init(){

SnakeValue=()

DrawBox

Values

X=$SnakeX

Y=$SnakeY

Snake $X $Y

echo -ne "\033[$Y;${X}H$SnakeHead"

values[`CordToKey $X $Y`]="snake"

echo -ne "\033[$ScoreY;${ScoreX}H \033[32m Score: $score \033[0m"

NewP

NewP

NewP

}

function MoveXYExit(){

local y

((y=PadY+Height+2))

echo -e "\033[?25h\033[${y};0H"

echo

exit

}

function MoveSnakeExit(){

kill -30 $pid

stty $sTTY

MoveXYExit

}

function MoveSnake(){

local key sig

pid=$1

sTTY=`stty -g`

echo -ne "\033[?25l"

trap "MoveSnakeExit" INT TERM

while :

do

sig=0

read -s -n 1 key

[[ "$key" == "A" ]] && sig=26

[[ "$key" == "B" ]] && sig=27

[[ "$key" == "C" ]] && sig=28

[[ "$key" == "D" ]] && sig=29

[[ "$key" == "q" ]] && MoveSnakeExit

[ $sig -ne 0 ] && kill -$sig $pid

done

}

## Main ##

if [ "$1" == "MoveXY" ];then

MoveXY

else

bash $0 MoveXY &

MoveSnake $! 2>/dev/null

fi