REVIEW OF COMPUTER ENGINEERING STUDIES
Vol.2, No.2, 2015, pp.5-8
http://dx.doi.org/10.18280/rces.020202

BASED ON THE NINE SUB-LOCK BEETLES FLASH GAME DESIGN

Kunliang Xu[1*]and Huimei Yan[2]

*[1,2] School of Computer Science and Engineering, Qujing Normal University, Sanjiang Road, Qilin District, Qujing, Yunnan, China.

Email: xkunl_2006@163.com

ABSTRACT

Flash software, game design based on an ancient network of small game, after the game starts, there will be a 15 × 15 checkerboard, the central board has a beetle, board the following nine pieces pieces, each falling one player chess pieces, beetles will move to no piece of lattice until it is surrounded by so far. The game is easy to operate, no need to install, the file size is small, suitable for network online.

Keywords: FLASH, Chessboard, Game.

I. INTRODUCTION

Because the FLASH game is easy to operate, green, no installation, small file size, etc., it is the majority of users' love [1]. FLASH game is basically the same as in the game's performance with the traditional form of the game, because of its small size, spread fast, beautiful screen, allowing the user's browser to install the ADOBE Flash player can play all Flash game, so much to replace trend traditional web online games. Thus, domestic production of unprovoked online with FLASH has become a trend [2].

In this paper, the design of the game's protagonist is a beetle. This is an old game. Legendary support four pillars have a day off, so it sent a fairy Jade Emperor to repair, he used nine stones will hold the seaside one beetles, used to make the cornerstone of broken columns, from the day after to stabilize, so this game is handed down. After the start of the game, a 15 × 15 checkerboard appears on the stage, there is a central board of beetles, board the following nine pieces pawn, a pawn players each fall, beetles will move to no piece of lattice until the siege live so far. Players take up to nine pieces surrounded by the victory of the beetles, if a player runs out of nine pieces, beetles have not been surrounded, the game failed [3-4].

2. FLASH SOFTWARE DESCRIPTION

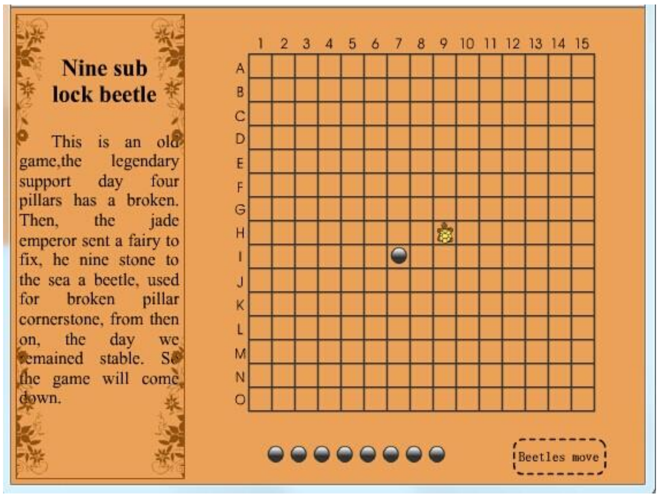
Flash, formerly known as Future Wave's Future Splash, is the world's first commercially available two-dimensional vector animation software for designing and editing Flash document. In November 1996, the U.S. company acquired Macromedia Future Wave, and renamed Flash. Macromedia was acquired by Adobe after another company. The latest version of Adobe Flash CS5 [5-6].

Flash is a vector-based graphics system, the elements are vector, just use a small amount of vector data can describe a complex object, graphics storage space is only a few thousandths of a bitmap and graphic zoom in and out and without reducing picture quality. It uses plug-in mode. Users simply install a plug-in, then you can quickly start and watch the animation. Supports bitmap, sound, gradient color, Alpha, transparency, etc., to support the network in real-time playback, very suitable for transmission over the network and online [5-6].

Flash's broad range of applications, in movies, television, cartoons, vocal music and other fields have been very good application, saving costs while ensuring the smooth and picture quality, and promote the integration and development of traditional media and the Internet media . With the further development of network technology, FLASH will be in application development, software interface development, the development of mobile phones in the field of game development, Web application services, site construction, media and entertainment, education system has been well applied, will be “the most flexible front desk.” [5-6].

3 .GAME INTERFACE DESIGN

The main game interface Game, board, beetles, beetles move several components material, the success or failure of voice prompts and material control buttons, the game runs the interface shown in Figure 1.


Nine sub lock beetle

This is an old game,the legendary support day four pillars has a broken. Then, the jade emperor sent a fairy to fix, he nine stone to the sea a beetle, used for broken pillar cornerstone, from then on, the day we remained stable. So the game will come down.

Figure 1. Game running interface map

Component design specific steps involved in the game are as follows [7-10]:

(1) New movie clip symbol named “Game”, draw a rectangle border, draw some pattern on the inside of the frame, insert static text on the inside “Nine sub lock beetle”, as well as a brief game text.

(2) New movie clip symbol named “chessboard”, draw a square and a square divided into 15 × 15 grid, each grid side length is 20 square above the input line number “1-15” on behalf of chessboard abscissa, enter the letters in the left side of the square “AO”, on behalf of the board ordinate.

(3) New movie clip symbol named “Herbie”, the layer name to “Herbie”, draw a slightly smaller than the checkerboard on the “Herbie” in the first 7,13,19 insert key frame in the first frame insert a blank keyframe 4,10,16 form beetles flashing animation. Create a new layer “actions”, in frame 19 to insert a keyframe on the frame to add the code “stop ()”.

(4) New button element, named “Hide button” in the button symbol “click” insert key frame, slightly smaller than the grid to draw a circle on the board.

(5) New movie clip symbol named “Hide button MC”, the button element “Hide button” into the first frame.

(6) New movie clip symbol named “beetles move”, the layer name to “bottom”, with a dotted line to draw a rounded rectangle not filled with color, insert the key in the first five frames, rounded rectangle to fill yellow. Create a new layer, named “Herbie Mobile”, enter the text “beetles move” on the rounded rectangle. New Layer “actions”, the first one to add code to “stop ();”. When the beetles move, this component starts playing.

(7) New button element named “Start.”

(8) New button element named “re-play.”

(9) new movie clip symbol named “result”, the layer name to “Results”, insert the key in the first two frames, draw a rounded rectangle, and enter text in the above “failure” in the first three insert key frame, the text “failure” to “success.” New Layer “actions”, respectively, in the first frame 1, 2, 3 add code “stop ();”.

4. THE GAME MAIN PROGRAM CODE

Beetles move the code below:

```
If (_root.playnow == “computer”) { // If you allow computer operation
xz = _root.xxx;
yz = _root.yyy;
dong = 0; // Determine in which direction it should move a pawn variables
_root.table[yz][xz] = 0;
dongArray = new Array(1, 2, 3, 4);
long = dongArray.length;
If (_root.table[yz-1][xz] == 1) {
dongArray[0] = 0;}// If the top of the turtle has been a pawn, the first element of the array dongArray of a 1 to 0
If (_root.table[yz+1][xz] == 1) {
    dongArray[1] = 0;
    }// If the bottom of the turtle has been a pawn, the first two elements of the array dongArray 2 to 0
If (_root.table[yz][xz-1] == 1) {
    dongArray[2] = 0;
// If the turtle has been a pawn of the left, the first three elements of the array dongArray of 3 to 0
    }
If (_root.table[yz][xz+1] == 1) {
    dongArray[3] = 0;
// If the turtle has been a pawn of the right, the first four elements of the array dongArray 4 to 0     }
For (num1=3; num1>0; num1--) {
    for (i=0; i<=num1; i++) {
        If (dongArray[i] == 0) {
            dongArray.splice(i, 1);
            long--;
            break;
            }
    }
}
If (_root.table[yz-1][xz-1] == 1 && _root.table[yz-1][xz+1] == 1 && _root.table[yz-2][xz] == 1 && long>1) {
    // If the beetles above vacancy has been surrounded by three pieces, and there are other spaces around beetles
    For (i=0; i<=3; i++) {
        If (dongArray[i] == 1) {
            dongArray.splice(i, 1);
            long--;
            break;
            }
    }
}
    // If the beetles above vacancy has been surrounded by three pieces, and there are other spaces around the child, beetles will not jump above grid
If (_root.table[yz+1][xz-1] == 1 && _root.table[yz+1][xz+1] == 1 && _root.table[yz+2][xz] == 1 && long>1) {
  // If the beetles following vacancy has been surrounded by three pieces, and there are other spaces around beetles
For (i=0; i<=3; i++) {
    If (dongArray[i] == 2) {
        dongArray.splice(i, 1);
        long--;
        break;
        }
    }
}
If (_root.table[yz-1][xz-1] == 1 && _root.table[yz+1][xz-1] == 1 && _root.table[yz][xz-2] == 1 && long>1) {
    // If a space has been left of the sub beetles surrounded by three pieces, and there are other spaces around beetles
    For (i=0; i<=3; i++) {
        If (dongArray[i] == 3) {
            dongArray.splice(i, 1);
            long--;
            break;
            }
    }
}
If (_root.table[yz-1][xz+1] == 1 && _root.table[yz+1][xz+1] == 1 && _root.table[yz][xz+2] == 1 && long>1) {
// If the beetles to the right of the space has been surrounded by three sub-pieces, and there are other spaces around beetles
  For (i=0; i<=3; i++) {
     If (dongArray[i] == 4) {
        dongArray.splice(i, 1);
        long--;
        break;
        }
  }
}
// In order to make the computer has become more "smart", not beetles move to have three pieces surrounded grid, unless you have nowhere to go.
suiji = random(long);
dong = dongArray[suiji];
If (dong == 1) {
    _root.gui._y -= 20;
    yz = yz-1;   }
If (dong == 2) {
    _root.gui._y += 20;
    yz = yz+1;
// If the dong is equal to 2, turtle walking down the grid value plus a longitudinal beetles
    }
    If (dong == 3) {
       _root.gui._x -= 20;
       xz = xz-1;
// If the dong is equal to 3, turtles left, beetles horizontal grid value minus one        }
    If (dong == 4) {
       _root.gui._x += 20;
       xz = xz+1;
// If dong equals four, turtle right away, beetles horizontal grid value plus a       }
    win = 0;
    If (_root.table[yz][xz+1] == 1 && _root.table[yz][xz-1] == 1 && _root.table[yz+1][xz] == 1 && _root.table[yz-1][xz] == 1) {
       win = 1;
    }
// If the upper and lower beetles have a piece about turtles have been around, the variable becomes a win
}
_root.xxx = xz;
_root.yyy = yz;
  guinum = yz*15+xz;// Georgia has made several beetles currently located and assigned to guinum
_root["dian"+_root.ti]._visible = 100; // Beetles walked where you can put the pieces
_root.ti = guinum;// After the change of the position beetles assigned to _root.ti
_root["dian"+guinum]._visible = 0 ;// Beetles can not recapture pawn go places
_root.playnow = "player";// In other player begins operation
If (_root.step == 9 && win == 0) {      // If you run out of nine full-pawn is not surrounded by beetles
    _root.jieguo.gotoAndPlay(2);
    _root.losesound.start();
    for (num=0; num<225; num++) {
        removeMovieClip(_root["dian"+num]);
    }      // Delete all the hidden buttons on the board, can not recapture pieces
}
If (win == 1) {    // If you win equals a beetle that has been around
    _root.jieguo.gotoAndPlay(3);
    _root.winsound.start();
    for (num=0; num<225; num++) {
        removeMovieClip(_root["dian"+num]);
    }      // Delete all the hidden buttons on the board, can not recapture pieces
}.
```

5. CONCLUSIONS

Game there are many flaws and shortcomings, such as the rough interface, not fine, level design is simple, challenging not high. But this game is designed for university learning outcomes me an important test, encountered many difficulties in the development process, but when solving a difficult time, and my heart will feel very happy, and also in the development process to get a difficult experience can be valuable.

REFERENCES

1. Qigui Fang, Flash Multimedia Courseware Case Tutorial [M], Beijing: Tsinghua University Press, 2012:60-80.
2. Weihua Zhu, Flash Animation Design Advantages Resolved [J], *Industry and Technology Forum*, 2013, (8):30-35.
3. Haihua Liu, Flash Game Development [J], Computer Software and Applications CD, 2011, (17):15-20.
4. Ke Liang, Xingying Fu, Yuan Liu, Flash Multimedia Courseware Design Study Works of Interactive Technology [J], *Software Tribune*, 2014, (2):20-30.
5. Xue Fang, Flash CS5 Training Tutorial Animation Case [M], Xi'an: Northwestern University Press, 2011:40-50.
6. Song Hu, Xiaokang Liang, Juan Zhao, Chinese Version of the Standard Flash CS5 Tutorial [M], Beijing: China Youth Press, 2011:20-35.
7. Yong Li, Wei Li, Zhenglin Song, Falsh Interactive Game Production Paradigm Navigation [M], Beijing: Tsinghua University Press, 2008:167-215.
8. Jingjing Chen, Talking about the Process of Teaching

Flash Animation Game Created [J], *Heilongjiang Science and Technology Information*, 2012, (8):50-55.

9. Zhao Zhi, Lu Zhao, Color Composition and Graphic Design Art [M], Liaoning: Fine Arts Publishing House, 1999:15-20.

10. Hao Yu, "Happy study group of words" Flash Game Development and Production [D], Wuhan: Wuhan University, 2011.