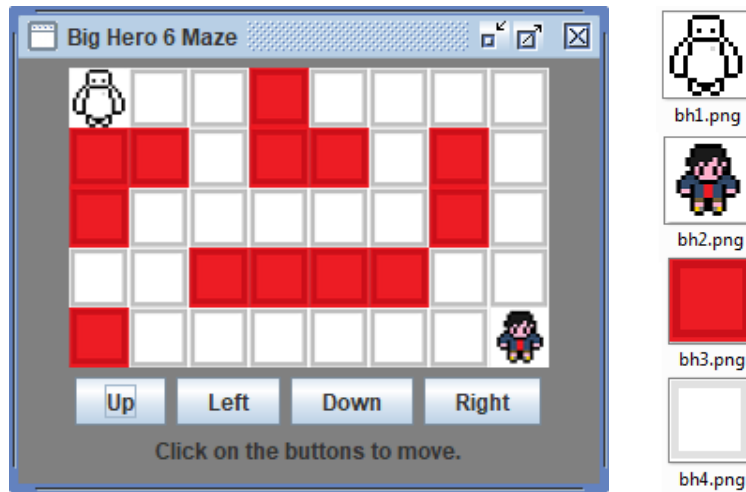


Repli.it Version of the Maze



Main.java

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.border.*;
```

```
public class Main {
    public static void main (String[] args)
    {
        JFrame.setDefaultLookAndFeelDecorated (true);
        //Create and set up the window.
        JFrame frame = new JFrame ("Big Hero 6 Maze");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        //Create and set up the content pane.
        JComponent newContentPane = new BigHero6 ();
        newContentPane.setOpaque (true);
        frame.setContentPane (newContentPane);
        frame.setSize (320, 260);
        frame.setVisible (true);
    }
}
```

BigHero.java

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.border.*;
//Purpose: To make a maze for the character to walk around

public class BigHero6 extends JPanel implements ActionListener
{
    //Note: Baymax (1) is NOT in the array
    int hero[] [] = {{4, 4, 4, 3, 4, 4, 4, 4},
                    {3, 3, 4, 3, 3, 4, 3, 4},
                    {3, 4, 4, 4, 4, 4, 3, 4},
                    {4, 4, 3, 3, 3, 3, 4, 4},
                    {3, 4, 4, 4, 4, 4, 4, 2}};

    int row = 5;
    int col = 8;
    JLabel pics[] = new JLabel [row * col];
    JLabel feedback;

    //Where did Baymax begin?
    int x = 0;
    int y = 0;
```

```

public BigHero6 ()
{
    setBackground (Color.gray);
    //code to build the screen
    Panel grid = new Panel (new GridLayout (row, col));
    int m = 0;
    for (int i = 0 ; i < row ; i++)
    {
        for (int j = 0 ; j < col ; j++)
        {
            pics [m] = new JLabel (createImageIcon ("bh" + hero [i] [j] + ".png"));
            pics [m].setPreferredSize (new Dimension (32, 32));
            grid.add (pics [m]);
            m++;
        }
    }
    add (grid);
    //Movement buttons
    JButton up = new JButton ("Up");
    up.setActionCommand ("up");
    up.addActionListener (this);
    add (up);
    JButton left = new JButton ("Left");
    left.setActionCommand ("left");
    left.addActionListener (this);
    add (left);
    JButton down = new JButton ("Down");
    down.setActionCommand ("down");
    down.addActionListener (this);
    add (down);
    JButton right = new JButton ("Right");
    right.setActionCommand ("right");
    right.addActionListener (this);
    add (right);
    feedback= new JLabel("Click on the buttons to move.");
    add(feedback);
    //move Baymax to the start position
    pics [x * col + y].setIcon (createImageIcon ("bh1.png"));
}

public void actionPerformed (ActionEvent e)
{
    //Up direction -----
    if (e.getActionCommand ().equals ("up"))
    {
        if (x - 1 < 0)
            feedback.setText ("Off the board!!");
        else if (hero [x - 1] [y] != 4)
            feedback.setText ("There is a wall.");
        else
        {
            feedback.setText ("OK! Moving up.");
            pics [x * col + y].setIcon (createImageIcon ("bh4.png"));
            x--;
            pics [x * col + y].setIcon (createImageIcon ("bh1.png"));
        }
    }
    //Down direction -----
    else if (e.getActionCommand ().equals ("down"))
    {
        if (x +1 >= row)
            feedback.setText ("Off the board!!");
        else if (hero [x+1] [y] != 4)
            feedback.setText ("There is a wall.");
        else
        {
            feedback.setText ("OK! Moving down.");
            pics [x * col + y].setIcon (createImageIcon ("bh4.png"));
            x++;
            pics [x * col + y].setIcon (createImageIcon ("bh1.png"));
        }
    }
    //Right direction -----
    else if (e.getActionCommand ().equals ("right"))

```

```

{
    if (y + 1 >= col)
        feedback.setText ("Off the board!!");
    else if (hero [x] [y + 1] != 4)
        feedback.setText ("There is a wall.");
    else
    {
        feedback.setText ("OK! Moving right.");
        pics [x * col + y].setIcon (createImageIcon ("bh4.png"));
        y++;
        pics [x * col + y].setIcon (createImageIcon ("bh1.png"));
    }
}
//Left direction -----
else if (e.getActionCommand ().equals ("left"))
{
    if (y-1 < 0)
        feedback.setText ("Off the board!!");
    else if (hero [x] [y-1] != 4)
        feedback.setText ("There is a wall.");
    else
    {
        feedback.setText ("OK! Moving left.");
        pics [x * col + y].setIcon (createImageIcon ("bh4.png"));
        y--;
        pics [x * col + y].setIcon (createImageIcon ("bh1.png"));
    }
}
}

protected static ImageIcon createImageIcon (String path)
{
    java.net.URL imgURL = BigHero6.class.getResource (path);
    if (imgURL != null)
        return new ImageIcon (imgURL);
    else
        return null;
}
}

```