

CS 205 - Programming for the Sciences Spring 2008 - Practice Exam 2 Answers

Problem 1

Frequency bar graph

```
private void frequency_Click(object sender, EventArgs e)
{
    int [] ranges = new int [10]; // 0-9, 10-19, etc.
    for (int i = 0; i < NUM_SCORES; i++)
    {
        int rangeIndex = scores[i]/10;
        ranges[rangeIndex]++;
    }

    results.Items.Add("The frequency of test scores is:");
    for (int i = 9; i >= 0; i--)
    {
        string bar = "";
        for (int j = 0; j < ranges[i]; j++)
            bar = bar + '*';
        int low = i * 10, high = i * 10 + 9;
        results.Items.Add(low.ToString() + '-' + high
            + ": " + bar);
    }
}
```

Problem 2

Tic Tac Toe

```
private void CheckForWinner()
{
    string winner = " ";
    if (grid[0, 0].Text == grid[0, 1].Text
        && grid[0, 1].Text == grid[0, 2].Text // top row
        && grid[0,0].Text != " ")
        winner = grid[0, 0].Text;
    else if (grid[1, 0].Text == grid[1, 1].Text
        && grid[1, 1].Text == grid[1, 2].Text // middle row
        && grid[1,0].Text != " ")
        winner = grid[1, 0].Text;
    else if (grid[2, 0].Text == grid[2, 1].Text
        && grid[2, 1].Text == grid[2, 2].Text // bottom row
        && grid[2,0].Text != " ")
        winner = grid[2, 0].Text;
    else if (grid[0, 0].Text == grid[1, 0].Text
        && grid[1, 0].Text == grid[2, 0].Text // left column
        && grid[0,0].Text != " ")
```

```

        winner = grid[0, 0].Text;
    else if (grid[0, 1].Text == grid[1, 1].Text
        && grid[1, 1].Text == grid[2, 1].Text // middle column
        && grid[0,1].Text != " ")
        winner = grid[0, 1].Text;
    else if (grid[0, 2].Text == grid[1, 2].Text
        && grid[1, 2].Text == grid[2, 2].Text // right column
        && grid[0,2].Text != " ")
        winner = grid[0, 2].Text;
    else if (grid[0, 0].Text == grid[1, 1].Text
        && grid[1, 1].Text == grid[2, 2].Text // top-left diagonal
        && grid[0,0].Text != " ")
        winner = grid[0, 0].Text;
    else if (grid[0, 2].Text == grid[1, 1].Text
        && grid[1, 1].Text == grid[2, 0].Text // top-right diagonal
        && grid[0,2].Text != " ")
        winner = grid[0, 2].Text;
    if (winner != " ")
    {
        results.Items.Add("The winner is " + winner);
        gameOver = true;
    }
    else
        results.Items.Add("There is no winner");
}

```

Problem 3

Temperatures

In `Temperature.cs`

```

public Temperature Fahrenheit()
{
    double newDegrees;
    switch (scale)
    {
        case 'C':
            newDegrees = degrees * 1.8 + 32;
            break;
        case 'F': // already in this scale
            newDegrees = degrees;
            break;
        default:
            throw new Exception("Should not happen, invalid scale");
    }
    return new Temperature(newDegrees, 'F');
}

public Temperature Celsius()
{

```

```

double newDegrees;
switch (scale)
{
    case 'C': // already in this scale
        newDegrees = degrees;
        break;
    case 'F':
        newDegrees = (degrees - 32) / 1.8;
        break;
    default:
        throw new Exception("Should not happen, invalid scale");
}
return new Temperature(newDegrees, 'C');
}

public static bool operator ==(Temperature left, Temperature right)
{
    // see if same scale
    if (left.scale == right.scale)
        return left.degrees == right.degrees;
    // if not, need to "convert" right temp to left's scale
    double newDegrees;
    if (left.scale == 'C') // right is in 'F'
        newDegrees = (right.degrees - 32) / 1.8;
    else // left is 'F', right is 'C'
        newDegrees = right.degrees * 1.8 + 32;
    return left.degrees == newDegrees;
}

public static bool operator !=(Temperature left, Temperature right)
{
    // see if same scale
    if (left.scale == right.scale)
        return left.degrees != right.degrees;
    // if not, need to "convert" right temp to left's scale
    double newDegrees;
    if (left.scale == 'C') // right is in 'F'
        newDegrees = (right.degrees - 32) / 1.8;
    else // left is 'F', right is 'C'
        newDegrees = right.degrees * 1.8 + 32;
    return left.degrees != newDegrees;
}

```

In Form1.cs

```

private void degree_Click(object sender, EventArgs e)
{
    Temperature t = Temperature.Parse(aTemperature.Text);
    results.Items.Add("The degrees of " + t + " is " + t.Degrees);
}

```

```

}

private void scale_Click(object sender, EventArgs e)
{
    Temperature t = Temperature.Parse(aTemperature.Text);
    results.Items.Add("The scale of " + t + " is " + t.Scale);
}

private void celsius_Click(object sender, EventArgs e)
{
    Temperature t = Temperature.Parse(aTemperature.Text);
    results.Items.Add("The Celsius equivalent of " +
        t + " is " + t.Celsius());
}

private void fahrenheit_Click(object sender, EventArgs e)
{
    Temperature t = Temperature.Parse(aTemperature.Text);
    results.Items.Add("The Fahrenheit equivalent of " +
        t + " is " + t.Fahrenheit());
}

private void compare_Click(object sender, EventArgs e)
{
    Temperature left = Temperature.Parse(leftTemperature.Text),
        right = Temperature.Parse(rightTemperature.Text);
    string answer = leftTemperature.Text + ' ' + anOp.Text
        + ' ' + rightTemperature.Text;
    switch (anOp.Text)
    {
        case "==":
            if (left == right)
                answer = answer + " is true";
            else
                answer = answer + " is false";
            break;
        case "!=":
            if (left != right)
                answer = answer + " is true";
            else
                answer = answer + " is false";
            break;
        case "<":
            if (left < right)
                answer = answer + " is true";
            else
                answer = answer + " is false";
            break;
        case ">":

```

```
        if (left > right)
            answer = answer + " is true";
        else
            answer = answer + " is false";
        break;
    default:
        throw new Exception("Invalid operation");
    }
    results.Items.Add(answer);
}
```