**一、产品设计方案**

1. **产品目标**运用python绘制美丽的樱花树，以让用户直观感受樱花树从春到冬的四季变化

**2. 产品内容总策划**（1） 流程规划：确定产品主题 **→** 讨论项目主要功能、主体 部分 **→** 上网搜索相关资料 **→** 结合相关资料编写代码 **→** 测试产品并进行改进  
（2）开发日程表：

|  |  |
| --- | --- |
| 6.3~6.6 | 确定产品主题、分工 |
| 6.8~6.12 | 讨论项目主要功能、主体部分，开始撰写系统开发说明文件 |
| 6.13~6.14 | 上网搜索相关资料 |
| 6.15~6.19 | 结合相关资料编写代码 |
| 6.19~6.21 | 测试产品并进行改进，制作ppt |
| 6.22~6.26 | 完善系统开发说明文件，撰写个人课程设计报告，制作和完善ppt |

**二、产品实现方案**

1. **系统的主要功能**运用python绘制美丽的樱花树，以让用户直观感受樱花树从春到冬的四季变化
2. **关键技术和技术难点  
   （1）**本软件采用turtle库，隐藏画笔后，绘制樱花树和该季节对应的景色。其中樱花树的树枝的分叉的方向采用random函数控制，使其看上去更自然，更像一棵真正的树。其中把树干当作第一个树枝画，树干的长度设置为60，接下来树枝缩短的长度依然由random函数控制，画笔的方向的转角也由random函数控制，使整棵树非常自然，比较符合自然界中树的走势。  
   **（2）**在本程序中，樱花树上的花和叶子也利用绘制树枝的代码进行绘制，实现方法为设置分支结构，如果branch（树枝）的长度在3到12像素之间，当作花（或者树叶）绘制，设置颜色为花（叶子）的颜色即可，长度大于12像素的branch当作真正的树枝绘制，设置颜色为树干的颜色。本软件代码中对于绘制樱花树定义的函数，采用了递归的方法，直至branch长度小于3像素，停止樱花树的绘制。  
   **（3）**本软件为了实现季节转变时樱花树上樱花变成绿叶，再变成黄色的枯叶，最终变成积雪的效果，实际上重复了绘画樱花树的过程，只是修改了对应位置颜色的设置。由于turtle库绘画的过程是动态的，在软件中的效果展示为樱花逐渐变成绿叶，绿叶逐渐变黄，黄色的叶子变成积雪的过程。  
   **（4）**本软件对于地上掉落的樱花，也采用了random函数实现随机掉落的效果。地上的樱花在夏天消失，也采用了与樱花树叶子和花变化的原理，用背景色对地上的樱花进行了覆盖。  
   **（5）**本软件还绘制了燕子、青蛙、稻草堆和雪人这些不同季节的景物，依然使用turtle绘制。代码较为简单，不予以赘述。  
   **（6）** 燕子、青蛙、稻草堆和雪人的消失也采用了覆盖绘制的原理，用背景色重新绘制了这些景物，制造出这些景物消失的效果。
3. **已完成的改进和存在的问题  
   （1）**由于random的随机导致每次生成的樱花树不一样，我们采用固定随机种子来控制树的整体形状。  
   **（2）**避免树的颜色太过单一，不同长度范围内的树叶设置成不同颜色。  
   **（3）**绘画樱花树以外事物的水平较低，导致出来的成品与意境较为不符。  
   **（4）**每个季节衔接比较生硬，事物消失的不够自然。