Wentworth Institute of Technology
COMP3071 – Intro To Programming Languages
Fall 2016, Kreimendahl

Programming Assignment 3

This assignment is intended to work with Piet, a simple visual programming language. It is inspired by Piet Mondrian's abstract paintings.

1. Repository

First, fork the repository at `https://gitlab.com/comp3071fk/pa3` and use git to clone the repository to your local machine. When you have completed the assignment, go through the git add/git commit/git push sequence to copy your files to your gitlab repository. Add me to the project on gitlab as a member with developer access so I have access to your code.

2. Specifications

The purpose of this assignment is to introduce a new language that is depicted rather differently than other languages we have studied so far.

2.1. Language

This assignment should be completed in Piet. Since you can't add comments directly to your code, you will attach a `readme.txt` file that has comments about your code.

2.2. Introduction

There are three important assignments for this assignment:

Language specification: http://www.dangermouse.net/esoteric/piet.html

Tutorial/reference: http://homepages.vub.ac.be/~diddesen/piet/index.html

IDE/debugger: http://www.rapapaing.com/blog/?page_id=6

I recommend reading through the specification and tutorial and trying to work a few example problems by hand to get a better sense of the language. Make sure you're familiar with the operations included with the language.

2.3. Problems

For this assignment, you will solve two problems using Piet. Each solution will be a png file, and you will add additional information for both in the same text file.

Odds:
This program takes a single input: the number of odd numbers to print. It should output that number of odd numbers, starting with 1. For example, if someone input 7, your program should output 1 3 5 7 9 11 13. This program should be turned in as `odds.png`.

Greatest Common Divisor:

This program takes two inputs: two numbers. It should output the greatest common divisor of the two inputs, and it should calculate it using the following formula (which was first written down about 2300 years ago):

```
gcd(a,b)
    while b≠0
        t:=b;
        b:=a mod b;
        a:=t;
    return a;
```

For example, inputs of 22 and 11 should output 11, and inputs of 1518 and 1428 should output 6 (since 6 is the largest number that divides evenly into both). This program should be turned in as `gcd.png`.

2.4. IDE/Submission

In the IDE, there are several important sections to note. First of all, the default canvas size is 10x10 and default codel size is also 10. You can change these in the Tools box. To save a program, there's a Save button in the File box. This will bring you to a new window that just has the image, presented as a php file by default. In your web browser, you can save your program by saving the webpage as `progName.png` – that's the proper file extension.

If you want to reload a saved program, you can upload the image file along with specifying the codel size. I recommend keeping the codel size at 10 to make it easier to remember.

Submission

For your final submission, include three files: your two programs (`odds.png` and `gcd.png`) and a `readme.txt` file that includes the following information:

1. A quick description of what each program does, like you would comment on any code block
2. A list of written commands (as described in the commands section of the spec) for each program. This allows for partial credit and is also a good intermediate step before you start drawing your actual program.

Please submit your png files with a **codel size of 10**.