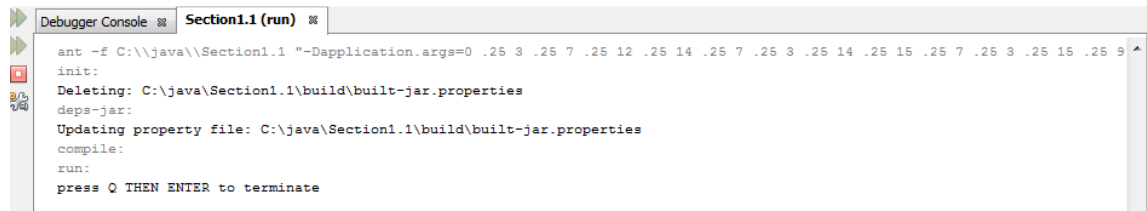


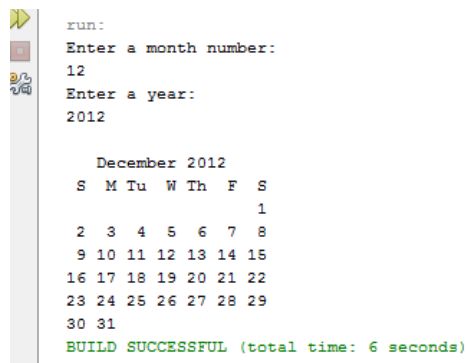
## Section 3

1. In the site notes, you will find the code for [PlayThatTuneDeluxe.java](#). It reads a text file sample given there to play a tune (For example, this is the file for [StairwayToHeaven.txt](#)). You are required to compile it and play that song. Once done that, modify the code to widen the spectrum to include new tones  $1/8$ ,  $1/4$  frequency harmonics. Now it only has  $1/2$ ,  $1$  and  $2$  tones. Assume equal weights of  $0.5$ .



```
ant -f C:\java\Section1.1 "-Dapplication.args=0 .25 3 .25 7 .25 12 .25 14 .25 7 .25 3 .25 14 .25 15 .25 7 .25 3 .25 15 .25 9"
init:
Deleting: C:\java\Section1.1\build\build-jar.properties
deps-jar:
Updating property file: C:\java\Section1.1\build\build-jar.properties
compile:
run:
press Q THEN ENTER to terminate
```

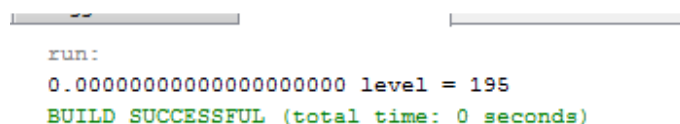
2. Modify the [Calendar.java](#) program that takes two command-line arguments  $m$  and  $y$  and prints out the monthly calendar for the  $m$ th month of year  $y$ .



```
run:
Enter a month number:
12
Enter a year:
2012

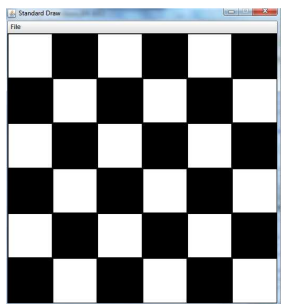
December 2012
S M Tu W Th F S
                1
2 3 4 5 6 7 8
9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31
BUILD SUCCESSFUL (total time: 6 seconds)
```

3. In the previous activity we have drawn the Sierpinski triangle using [Sierpinski.java](#). We know from elementary school how to compute the area of a triangle. Can you come up with a static class to compute the black area of a Sierpinski Triangle? (white is the excluded area).



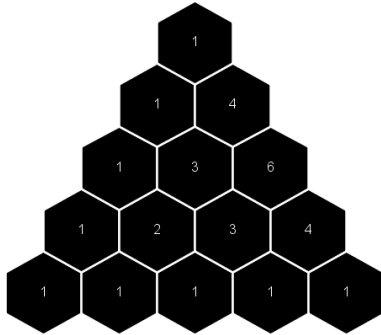
```
run:
0.00000000000000000000000000000000 level = 195
BUILD SUCCESSFUL (total time: 0 seconds)
```

4. Modify the code [DiamondTile.java](#) to produce a black and light colored chess board. You can consult the [Class StdDraw](#) to guide you thru this exercise.



### Section 3

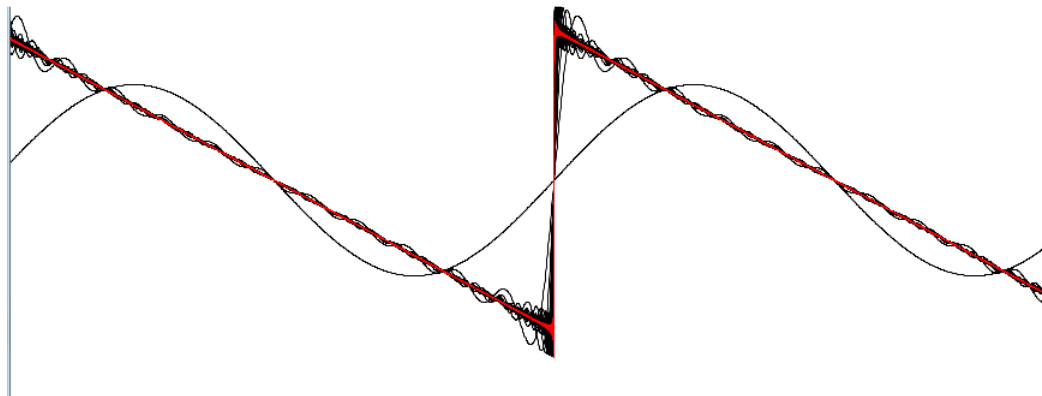
- Run the code [HexTile.java](#) to produce an hexagonal tile array. Now, modify the code to construct a Pascal triangular pyramid showing the values in the center of the hexagons. You have used [Pascal.java](#) code in Activity 2.



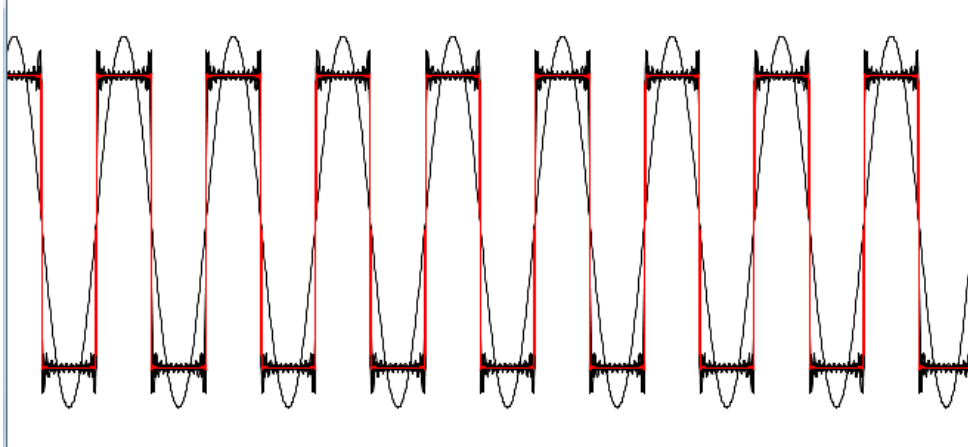
- Write a Java static class code for checking the validity of a Visa or MasterCard credit card, using the [IBM Digit Verification](#) description.

```
run:
Enter a card number:
12347548429999
Invalid number!!
BUILD SUCCESSFUL (total time: 6 seconds)
|
```

- Write a java class for approximating a [square wave](#) by a sum of N sinusoidal functions:  $\frac{4}{\pi} [\sin(1 \cdot 2\pi t)/1 + \sin(3 \cdot 2\pi t)/3 + \sin(5 \cdot 2\pi t)/5 + \dots]$ , where  $\pi=3.1415\dots$ . Plot it as a function of time t, and see how this sum approximates the square wave when adding more terms. This is known as Fourier Sine Series transformation. Then play the approximation using the StdAudio java class. You can follow the [SawTooth.java](#) example to approximate a [sawtooth wave](#) signal.



### Section 3



8. You are invited to use the [iterated function system IFS.java](#). This code plots an iteration algorithm for the different data. Please reproduce the following figures starting from input files: [barnsley.txt](#), [sierpinski.txt](#), [tree.txt](#). Play with the files in order to reproduce different figures, different colors, etc. Produce 3 figures you like. Remember that you need a large number of points to visualize a figure, like  $N=10000$  at least. You can read more about these figures: [Barnsley fern](#), [Sierpinski triangle](#), [Fractal trees](#).

