Computer Science 327
Project 2, Part a
C++ Programming Project
Digital Sound Cont.

The Assignment (Part a)

In this part of the assignment you will add two features to the SoundSamples class that will make for better sounds.

1. (100 points) Add a new and different reverb function to the SoundSamples class named "reverb2" This method takes the same parameters as the old reverb function, but computes the new wave in an slightly different way. The protype for this function is:

   ```
   void reverb2( float delay, float attenuation);
   ```

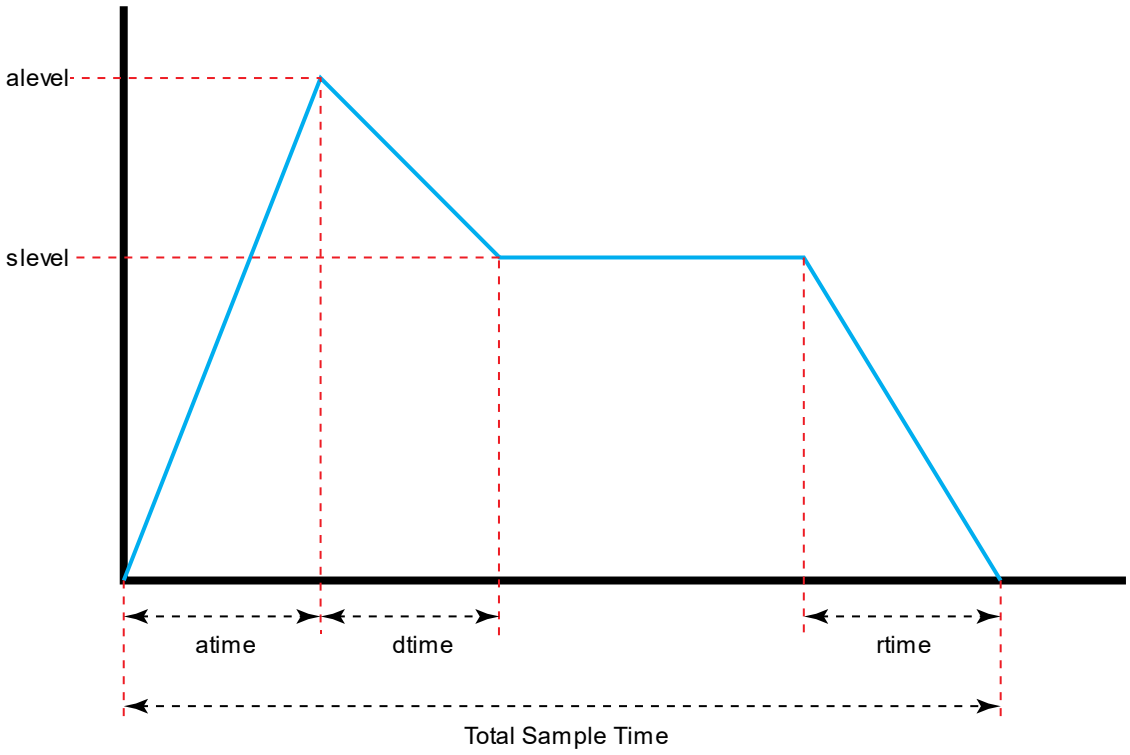
   With the idea that this sound processing function operates on the samples stored in "this" object. Unlike the old reverb function, you will not be using the filter function to do this version. The difference is that you will be adding a delayed sample, but the delayed sample will be one that you have already generated. The following example should make this clear.

   Let x be the original array of samples currently stored in this object, and let y be a new temporary array of samples. Then y[i] = x[i] + y[i-sdelay], where sdelay is the number of samples (to the nearest int) that represents the delay parameter given the sample rate for the sequence. Note that if i-sdelay is less than zero, then y[i] = x[i]; do not add in any delayed value. Valid values for a delay is any number greater or equal to zero. Valid values for an attenuation is any number greater or equal to zero. Note that combinations of higher attenuation values and specific delays could cause positive feedback. You probably have heard this when a microphone is placed close to a speaker. After y is generated as in above, it replaces the old sampled sequence stored in the object.

2. (200 points) Add an ADSR function to the SoundSamples class. This is similar to the reverb2 function in the sense that it modifies the current sequences of samples in the object. ADSR stands for Attack, Decay, Sustain, and Release, and they represent different times in the wave. If you google ADSR you will find lots of information about how to define these things. Please use the one I've defined here. The function prototype you are to add is:

```
adsr( float atime, alevel, dtime, slevel, rtime);
```

Where atime, alevel, dtime, slevel, rtime are defined by the graph below.



The way this works is to sample by sample multiple the samples of the sound sequence by the corresponding value of the ADSR graph. Typically, alevel and slevel is less than or equal to 1, but they could be greater. There are several possible error conditions which can be handled by your preference. Most of these occur when the atime, dtime and rtime sum to a number greater than the total sample time.

3. (50 points) Write a main function and place it in the file "TestMain.cpp" The main function asks the user for a wave type followed by reverb and ADSR parameters. The program then asks the user for a filename to place the output followed by any number of note numbers terminated with an input of a negative number. The program then writes each note using the specified wave type and parameters given by the user at the frequency corresponding to the note number to the output file. Each note is a duration of 1 second with 0.25 seconds between each note.