My additional goal was to simulate the expansion and compression of an ideal gas. A large number of identical particles were placed in a 20x20 box at random positions with random velocities. The right-hand wall is moved towards and away from left-hand wall at constant speed. The simulation runs for 20 seconds. A plot of the internal energy of the gas against volume (or area, since it is a two-dimensional system) can be made.

The program creates the system using two structs: one for the particles, and one for the walls. The walls are stored in one object, and the particles in an array of objects. Since a large number of particles are used, the memory was allocated before the data was set. Functions were used to calculate the new positions of the particles, and the new velocities if any collisions occur. To reduce the runtime of the program, the particles only check if they have collided with particles before them in the array. The right-hand wall was also moved using functions. The program was split into three files: ‘bonkers.cc’, which contains the main function, ‘functions.cc’, which contains the definitions of all functions used in the program, and ‘bonkers.h’, which contains global constants, function declarations and structure declarations. A makefile was used to compile the programs. Four gnuplot scripts have been used too: ‘bonkers.gnu’ and ‘plot.gnu’, which create the frames for the animation; ‘energy.gnu’, which plots the energy-volume graph, and ‘energy2.gnu’; which plots the energy-1/volume graph.

The system being simulated is an ideal gas that is undergoing a change in volume over time. The ideal gas consists of a large number (1000) circles of identical radius and mass. At the start of the program the particles are placed at random positions and with random velocities. The particles are then moved to their new projected position after constant time intervals of 0.01 s. If two particles collide, then their new velocities are calculated, assuming all collisions are elastic. The particles can also collide with the walls, in which case the same procedure is followed: the new velocities of the particle and the wall are calculated. Since the walls have infinite mass, they do not change velocity, so in general the gas is unable to expand. The right-hand wall moves from x = 0 to x = 10, at constant speed, causing expansion and compression of the gas. When particles collide with this wall, they do work or work is done on them (depending on the motion of the wall). This causes the internal energy of the gas to increase as it is compressed and decrease as it expands, as expected from an ideal gas. This can be seen in the animation: at lower volumes the particles move faster, and at higher volumes the particles move slower. A graph of total energy against volume could also be plotted. Since no heat flows into or out of the system, the system undergoes adiabatic compression or expansion. This is characterised by *pV*γ = constant, where *p* is the pressure, *V* is the volume, and *γ* is the adiabatic constant. For a two-dimensional monatomic gas *γ* = 2 ((4*R*/2) / (2*R*/2)). The work done on the gas is given by *pV*, and so a plot of internal energy against *V*1-*γ*(here *V*-1)should give a straight line. The internal energy is calculated by adding up the kinetic energy (ideal gas, so no potential energy), of each particle.

The plot of internal energy against volume showed that as volume decreased, the internal energy increased in a non-linear curve. This is to be expected, as the internal energy is proportional to the pressure, so the curve should resemble an adiabatic. The plot of internal energy against 1/volume showed (ignoring the standard deviations) a straight-line plot (there is a slight curve, but this is likely due to the issue discussed later), confirming that this process is indeed adiabatic. The two plots exhibit a strange behaviour though, after each cycle the internal energy ends slightly higher. This is likely due to the wall expanding through its own process, not just by the gas pushing against it. This is equivalent to work being done on the wall by the surroundings. This additional work is left in the gas after the expansion is finished.

This project has helped me visualise how ideal gases move, and how the energy varies in an adiabatic system. It has also taught me how to deal with a large number of particles in a system. This will be useful in future years, when I need to simulate large, complex systems, to see what is happening.

My solution was in files named ‘bonkers.cc’, ‘functions.cc’, ‘bonkers.h’, ‘Makefile’, ‘bonkers.gnu’, ‘plot.gnu’,’energy.gnu’, and ‘energy2.gnu’.