UMEÅ UNIVERSITY
Department of Physics
Claude Dion

February 11, 2015

Numerical Methods in Physics
(5FY033)

Phonons in rare gases

PURPOSE: To learn how to use routines for solving eigenvalue problems.
To determine the dispersion relation for phonons in rare gas crystals.
To determine the temperature dependence of the heat capacity by using the dispersion relation.

LITERATURE: N. W. Ashcroft and N. D. Mermin, Solid State Physics (Saunders, New York, 1976)
M. Galassi *et al.*, GNU Scientific Library Reference Manual, 2nd ed. (Network Theory, Bristol, 2006), `http://www.gnu.org/software/gsl/`
W. H. Press *et al.*, Numerical Recipes in C, 2nd ed. (Cambridge University Press, Cambridge, 1992), chap. 11

1 Introduction

The atoms in a crystal have their equilibrium positions arranged in a periodic lattice, but the atoms will vibrate about these equilibrium positions. These vibrations can be described in terms of phonons. A phonon is an undulating motion in the crystal that is characterized by a frequency, a wavelength and a direction of polarization. The purpose of this exercise is to determine possible frequencies for a phonon with a certain wavelength. Knowledge of these frequencies enables us to calculate theoretically the heat capacity of the crystal.

We focus on crystals containing only one type of atom. Let us define every position in the lattice by $\mathbf{l} = (l_{x_1}, l_{x_2}, l_{x_3})$; this is a vector of integers, which tells the position of every lattice point measuring the length in terms of the lattice constant a. The atoms can be taken off their equilibrium position, and the displacement is given by $\mathbf{u}^l = (u^l_{x_1}, u^l_{x_2}, u^l_{x_3})$. The potential energy depends upon the relative distance of the atoms; considering an expansion until the second order, we obtain [1]

$$\begin{aligned} U \approx U_0 &+ [\text{vanishing first derivatives of } \phi] \\ &+ \frac{1}{2} \sum_{\mathbf{l},\mathbf{l}',\alpha,\beta} (u^l_\alpha - u^{l'}_\alpha) \frac{\partial^2 \phi(\mathbf{l}-\mathbf{l}')}{\partial x_\alpha \, \partial x_\beta} (u^l_\beta - u^{l'}_\beta), \end{aligned} \tag{1}$$

where the sum over $\mathbf{l}$ and $\mathbf{l}'$ is intended to count each pair of sites once, *i.e.*, if the sum already contains the term $\mathbf{l} = \mathbf{l_1}$ and $\mathbf{l}' = \mathbf{l_2}$, it does not contain the term $\mathbf{l} = \mathbf{l_2}$ and $\mathbf{l}' = \mathbf{l_1}$. Notice that U_0 is just a constant which does not contribute to the equations of motion. Moreover, the quadratic term can be rewritten as

$$U_{\text{harm}} = \frac{1}{2} \sum_{\mathbf{l},\mathbf{l}',\alpha,\beta} u^{\mathbf{l}}_\alpha D^{\mathbf{l}\mathbf{l}'}_{\alpha\beta} u^{\mathbf{l}'}_\beta, \tag{2}$$

where

$$D^{\mathbf{l}\mathbf{l}'}_{\alpha\beta} = \delta_{\mathbf{l}\mathbf{l}'} \sum_{\mathbf{l}''} \frac{\partial^2 \phi(\mathbf{l}-\mathbf{l}'')}{\partial x_\alpha \partial x_\beta} - \frac{\partial^2 \phi(\mathbf{l}-\mathbf{l}')}{\partial x_\alpha \partial x_\beta}. \tag{3}$$

Notice that (2) and (3) are just a way to rewrite the harmonic term of (1). With this notation the equations of motion are

$$m \ddot{u}^{\mathbf{l}}_\alpha = -\sum_{\mathbf{l}'\beta} D^{\mathbf{l}\mathbf{l}'}_{\alpha\beta} u^{\mathbf{l}'}_\beta \tag{4}$$

where α and β can be 1, 2, or 3. This equation of motion for atom l is used later to formulate the eigenvalue problem.

2 Model for rare gas crystals

At low temperatures, the rare gases (except He) condense into a face centered cubic lattice (see figure 1). We denote the length of a side in the conventional unit cell $2a$.

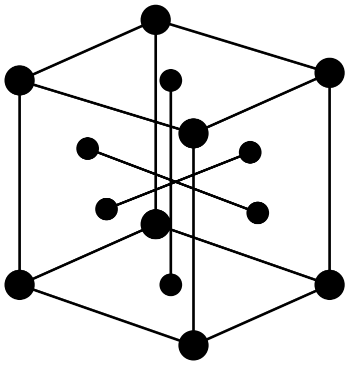

Figure 1: Lattice points in a face centered cubic lattice.

	Ne	Ar	Kr	Xe
σ (Å)	3.035	3.709	3.966	4.318
ϵ (10^{-20} J)	0.0721	0.236	0.325	0.458
r_{nn} (Å)	3.1562	3.7477	3.9922	4.3346
m (10^{-25} kg)	0.335 092	0.663 35	1.3915	2.180 17

Table 1: Model parameters for rare gas crystals. (1 Å = 10^{-10} m)

Every atom has 12 nearest neighbors. Placing the origin at the center of one of the atoms, the coordinates of its nearest neighbors are:

$(a, a, 0)$	$(a, -a, 0)$	$(-a, a, 0)$	$(-a, -a, 0)$	$(a, 0, a)$	$(a, 0, -a)$
$(-a, 0, a)$	$(-a, 0, -a)$	$(0, a, a)$	$(0, a, -a)$	$(0, -a, a)$	$(0, -a, -a)$.

We assume that the atoms affect each other through central forces, *i.e.*, the force (and the potential energy) depends only on the distance between two atoms. The potential energy between a pair of atoms is denoted $\Phi(r)$, where r is the distance between the two atoms. The total potential energy in the crystal then becomes a sum over $\Phi(r)$ for every pair of atoms. We further assume that an atom only interacts with its 12 nearest neighbors.

For the rare gases a special pair potential, the Mie-Lennard-Jones potential, is often used,

$$\Phi(r) = 2\epsilon \left[\frac{1}{2} \left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^{6} \right]. \tag{5}$$

Values for the parameters σ and ϵ for four rare gases are given in table 1.

In the equation of motion we need the second derivatives of the potential energy. We put $\mathbf{r} = (x_1, x_2, x_3)$ and define

$$B = \left[\frac{1}{r} \frac{\partial \Phi(r)}{\partial r} \right]_0 \equiv \frac{1}{r_{\mathrm{nn}}} \Phi'(r_{\mathrm{nn}}), \tag{6}$$

$$A = \left[\frac{\partial^2 \Phi(r)}{\partial r^2} \right]_0 \equiv \Phi''(r_{\mathrm{nn}}), \tag{7}$$

where r_{nn} is the distance between two nearest neighbors at their equilibrium positions. With these definitions we can write

$$\frac{\partial^2 \phi(\mathbf{l}-\mathbf{l}')}{\partial x_\alpha \partial x_\beta} = \frac{x_\alpha x_\beta}{x_1^2 + x_2^2 + x_3^2}[A - B] + \delta_{\alpha\beta} B. \tag{8}$$

where $x_1 = l_{x_1} - l'_{x_1}$, $x_2 = l_{x_2} - l'_{x_2}$, and $x_3 = l_{x_3} - l'_{x_3}$[1]; notice that in (8) if $\mathbf{l} - \mathbf{l}' = 0$ then $x_\alpha x_\beta = 0$, and the element proportional to $[A - B]$ vanishes.

3 Collecting the terms

Let us focus our attention on a definite atom located in (000), and let us suppose that the only interactions that count in the sum (4) are the "self-interaction" $\mathbf{l} = (000)$ and the interactions with all the 12 nearest neighbors $\mathbf{l}' = (1\,1\,0), (1\,\bar{1}\,0), (1\,0\,1), \ldots$ Let us now break the sum

$$-\sum_{\mathbf{l}'\beta} D_{\alpha\beta}^{\mathbf{l}\mathbf{l}'} u_\beta^{\mathbf{l}'} \tag{9}$$

into a few parts:

- **Case 1:** $\alpha = 1,\ \beta = 1,\ l = 000,\ l' = n.n.$

  In this case, due to the term $x_\alpha x_\beta = x_1^2$ in (8) there are 8 terms which have a non vanishing $[A - B]$ element

$\mathbf{l}'$	potential term
$(1\,1\,0)$	$\frac{1}{2}[A-B]u_{x_1}^{(1\,1\,0)} + Bu_{x_1}^{(1\,1\,0)}$
$(1\,\bar{1}\,0)$	$\frac{1}{2}[A-B]u_{x_1}^{(1\,\bar{1}\,0)} + Bu_{x_1}^{(1\,\bar{1}\,0)}$
$(\bar{1}\,\bar{1}\,0)$	$\frac{1}{2}[A-B]u_{x_1}^{(\bar{1}\,\bar{1}\,0)} + Bu_{x_1}^{(\bar{1}\,\bar{1}\,0)}$
$(1\,0\,1)$	$\frac{1}{2}[A-B]u_{x_1}^{(1\,0\,1)} + Bu_{x_1}^{(1\,0\,1)}$
$(1\,0\,\bar{1})$	$\frac{1}{2}[A-B]u_{x_1}^{(1\,0\,\bar{1})} + Bu_{x_1}^{(1\,0\,\bar{1})}$
$(\bar{1}\,0\,1)$	$\frac{1}{2}[A-B]u_{x_1}^{(\bar{1}\,0\,1)} + Bu_{x_1}^{(\bar{1}\,0\,1)}$
$(\bar{1}\,0\,\bar{1})$	$\frac{1}{2}[A-B]u_{x_1}^{(\bar{1}\,0\,\bar{1})} + Bu_{x_1}^{(\bar{1}\,0\,\bar{1})}$

  while there are other 4 term for which $x_\alpha = x_\beta = x_1 = 0$, so that the $[A - B]$ element does not appear:

[1] These lengths should be multiplied by a. However this constant appears both in the numerator and the denominator.

$\mathbf{l}'$	potential term
$(0\,1\,1)$	$Bu_{x_1}^{(0\,1\,1)}$
$(0\,1\,\bar{1})$	$Bu_{x_1}^{(0\,1\,\bar{1})}$
$(0\,\bar{1}\,1)$	$Bu_{x_1}^{(0\,\bar{1}\,1)}$
$(0\,\bar{1}\,\bar{1})$	$Bu_{x_1}^{(0\,\bar{1}\,\bar{1})}$

- **Case 2:** $\alpha = 1,\ \beta = 1,\ l = 000,\ l' = 000$

  In this case $x_1 = 0$, so that the only term that survives in (9) is

$$-\left[\sum_{\mathbf{l}''} \frac{\partial^2 \phi(\mathbf{l}-\mathbf{l}'')}{\partial x_1 \partial x_1} + B\right] u_x^{(0\,0\,0)} = -\sum_{\mathbf{l}''}\left\{\frac{x_1^2}{x_1^2+x_2^2+x_3^2}[A-B] + B\right\}$$
$$= -\frac{8}{2}[A-B]u_x^{(0\,0\,0)} - 13Bu_x^{(0\,0\,0)} + Bu_x^{(0\,0\,0)}$$
$$= -\frac{8}{2}[A-B]u_x^{(0\,0\,0)} - 12Bu_x^{(0\,0\,0)}. \qquad (10)$$

  The summation over $\mathbf{l}''$ has 13 elements, the atom plus the 12 nearest neighbors. The factor 8 in front of $[A-B]$ is due to the fact this term vanishes whenever the summation over $\mathbf{l}''$ has $l_{x_1} = 0$, so only 8 terms appear.

- **Case 3:** $\alpha = 1,\ \beta = 2,\ l = 000,\ l' = 000$

  In this case we have

$$-\sum_{\mathbf{l}'\beta} D_{\alpha\beta}^{\mathbf{l}\mathbf{l}'} u_\beta^{\mathbf{l}'} = -\sum_{\mathbf{l}''} \frac{\partial^2 \phi(\mathbf{l}-\mathbf{l}'')}{\partial x_1 \partial x_2} u_y^{(0\,0\,0)}$$
$$= -\sum_{\mathbf{l}''} \frac{x_1 x_2}{x_1^2+x_2^2+x_3^2}[A-B]u_y^{(0\,0\,0)}. \qquad (11)$$

  Notice that the term proportional to B is not present, because it is multiplied by $\delta_{12} = 0$. The sum in (11) is composed of the terms in which both x_1 and x_2 are non-vanishing, *i.e.*,

$\mathbf{l}''$	potential term
$(1\,1\,0)$	$+\frac{1}{2}[A-B]u_y^{(0\,0\,0)}$
$(\bar{1}\,1\,0)$	$-\frac{1}{2}[A-B]u_y^{(0\,0\,0)}$
$(\bar{1}\,\bar{1}\,0)$	$+\frac{1}{2}[A-B]u_y^{(0\,0\,0)}$
$(1\,\bar{1}\,0)$	$-\frac{1}{2}[A-B]u_y^{(0\,0\,0)}$

  The alternating sign is due to the factor $x_1 x_2$ in front of the term $[A-B]$, which causes *the sum to vanish.* No contribution to the total sum for u_y^{000}.

- **Case 4:** $\alpha = 1,\ \beta = 2,\ l = 000,\ l' = n.n.$

There are no terms proportional to B, due to the factor $\delta_{12} = 0$, and the sum is made by the term in which both x_1 and x_2 are not zero.

$\mathbf{l}'$	potential term
$(1\,1\,0)$	$+\frac{1}{2}[A-B]u_y^{(1\,1\,0)}$
$(1\,\bar{1}\,0)$	$-\frac{1}{2}[A-B]u_y^{(1\,\bar{1}\,0)}$
$(\bar{1}\,1\,0)$	$-\frac{1}{2}[A-B]u_y^{(\bar{1}\,1\,0)}$
$(\bar{1}\,\bar{1}\,0)$	$+\frac{1}{2}[A-B]u_y^{(\bar{1}\,\bar{1}\,0)}$

- **Case 5:** $\alpha = 1,\ \beta = 3,\ l = 000,\ l' = 000$

  Same reasoning as for $\beta = 2$.

- **Case 6:** $\alpha = 1,\ \beta = 3,\ l = 000,\ l' = n.n.$

  Same reasoning as for $\beta = 2$.

Summing all the terms in the equation of motion for $u_x^{(0\,0\,0)}$ becomes

$$\begin{aligned}
-m\omega^2 u_x^{(0\,0\,0)} = \frac{1}{2}\left[A-B\right] & \Big\{ u_x^{(1\,1\,0)} + u_x^{(1\,\bar{1}\,0)} + u_x^{(\bar{1}\,1\,0)} + u_x^{(\bar{1}\,\bar{1}\,0)} + u_x^{(1\,0\,1)} \\
& + u_x^{(1\,0\,\bar{1})} + u_x^{(\bar{1}\,0\,1)} + u_x^{(\bar{1}\,0\,\bar{1})} - 8u_x^{(0\,0\,0)} \Big\} \\
& + B \Big\{ u_x^{(1\,1\,0)} + u_x^{(1\,\bar{1}\,0)} + u_x^{(\bar{1}\,1\,0)} + u_x^{(\bar{1}\,\bar{1}\,0)} + u_x^{(1\,0\,1)} \\
& + u_x^{(1\,0\,\bar{1})} + u_x^{(\bar{1}\,0\,1)} + u_x^{(\bar{1}\,0\,\bar{1})} + u_x^{(0\,1\,1)} + u_x^{(0\,1\,\bar{1})} + u_x^{(0\,\bar{1}\,1)} \\
& + u_x^{(0\,\bar{1}\,1)} - 12u_x^{(0\,0\,0)} \Big\} \\
& + \frac{1}{2}\left[A-B\right] \Big\{ u_y^{(1\,1\,0)} - u_y^{(1\,\bar{1}\,0)} - u_y^{(\bar{1}\,1\,0)} + u_y^{(\bar{1}\,\bar{1}\,0)} \Big\} \\
& + \frac{1}{2}\left[A-B\right] \Big\{ u_z^{(1\,0\,1)} - u_z^{(1\,0\,\bar{1})} - u_z^{(\bar{1}\,0\,1)} + u_z^{(\bar{1}\,0\,\bar{1})} \Big\} . \qquad (12)
\end{aligned}$$

4 The dynamical matrix

In order to further simplify equation (12) we make the *ansatz* that the solution is a plane wave,

$$u_\alpha^l = \frac{\epsilon_\alpha}{\sqrt{m}} \mathrm{e}^{\mathrm{i}(\mathbf{k}\cdot\mathbf{r}_l - \omega t)}, \qquad (13)$$

where $\mathbf{r}_l = a\mathbf{l}$ is the equilibrium position for the atom l and the wave vector $\mathbf{k}$ has to be in the first Brillouin zone (see figure 2). In this way (12) is rewritten as

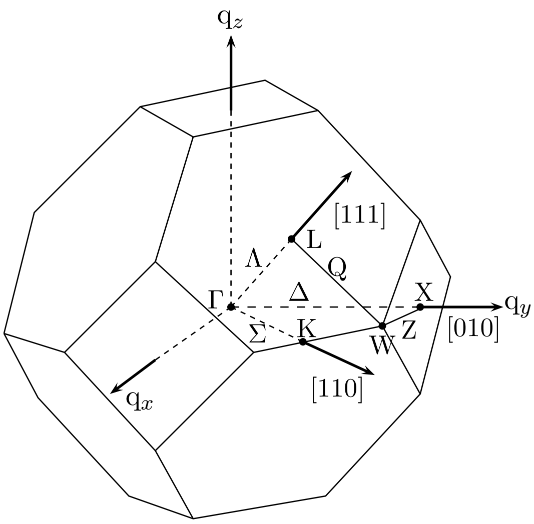

Figure 2: The first Brillouin zone for a fcc lattice.

$$\begin{aligned}
\omega^2 \varepsilon_x \quad = \quad & \left\{ \frac{1}{2m} [A + B] \left[8 - 4\cos(k_x a)\cos(k_y a) - 4\cos(k_x a)\cos(k_z a) \right] \right. \\
& \left. + \frac{B}{m} \left[4 - 4\cos(k_y a)\cos(k_z a) \right] \right\} \varepsilon_x \\
& + \frac{1}{2m} [A - B] \left[(4\sin(k_x a)\sin(k_y a) \right] \varepsilon_y \\
& + \frac{1}{2m} [A - B] \left[4\sin(k_x a)\sin(k_z a) \right] \varepsilon_z . \qquad (14)
\end{aligned}$$

This equation defines the first row in the dynamical matrix, $\hat{D}$, defined such that

$$\hat{D}(\mathbf{k})\hat{\varepsilon}(\mathbf{k}) = \omega^2(\mathbf{k})\hat{\varepsilon}(\mathbf{k}). \qquad (15)$$

By introducing

$$\mathbf{k} = \mathbf{q}\frac{\pi}{a}, \qquad (16)$$

the arguments in the trigonometric functions can be simplified, *e.g.*, $k_x a \rightarrow q_x \pi$.

Exercise 1:

Derive the two remaining rows in the dynamical matrix for one of the rare gases using *one* of the following methods:

1. Write down the equations of motion in the y- and z-directions and derive the equations corresponding to equation (12) and (14).

2. Note that the crystal looks the same in the x-, y- and z-directions. The remaining two rows in the dynamical matrix can be obtained by a suitable permutation of the indices x, y and z.

Exercise 2a:

Plot the dispersion relation $\omega(\mathbf{q})$ in the symmetry directions [(1 0 0), (1 1 0), and (1 1 1)].

Now we have a theoretical model from which we can also calculate how the frequencies are affected if we reduce the lattice constant, *e.g.*, by applying pressure on the crystal. As a first approximation, we assume that the constants σ and ϵ do not change with the pressure. It seems to be a correct assumption at least for small changes of the volume. The volume dependence of the phonon frequencies are often described by the quantity

$$\gamma_j(\mathbf{q}) = -\frac{\partial \ln \omega(\mathbf{q}, j)}{\partial \ln V}. \tag{17}$$

Exercise 2b:

Calculate $\gamma_j(\mathbf{q})$ for phonons in the symmetry directions. Use a finite difference approximation for the derivative, *i.e.*,

$$\frac{\mathrm{d}f(x)}{\mathrm{d}x} \simeq \frac{f(x+h) - f(x-h)}{2h}.$$

5 The heat capacity

One can derive the following expression for the heat capacity:

$$C_V = k_\mathrm{B} \sum_{\mathbf{k},j} \left[\frac{\hbar\omega(\mathbf{k},j)}{k_\mathrm{B}T}\right]^2 \frac{\exp\left[\frac{\hbar\omega(\mathbf{k},j)}{k_\mathrm{B}T}\right]}{\left(\exp\left[\frac{\hbar\omega(\mathbf{k},j)}{k_\mathrm{B}T}\right] - 1\right)^2}, \tag{18}$$

where k_B is the Boltzmann constant and j denotes a branch in the dispersion relation.

For macroscopically large crystals the discrete $\mathbf{k}$-vectors lie so closely spaced that we can replace the sum by an integral,

$$\sum_{\mathbf{k}} \rightarrow \frac{V}{(2\pi)^3} \int \mathrm{d}\mathbf{k}.$$

Due to symmetry we integrate over 1/48 part of the Brillouin zone and multiply the result by 48. To perform the numerical integration we approximate the integral by a sum over unit volumes, $\Delta\mathbf{k}$. Each unit volume is represented by a $\mathbf{q}$-vector. The value of the integrand is calculated for this **q**-vector and then multiplied by the unit volume. Finally, all contributions are summed:

$$\frac{C_V}{V} = \frac{1}{(2\pi)^3} \sum_j \sum_{\mathbf{q}} \Delta\mathbf{k}\, f_j(\mathbf{q}), \tag{19}$$

where $f_j(\mathbf{q})$ is obtained from eq. (18).

In the file `~fnm/rare/qvekt` there is a set of **q**-vectors that can be used. These vectors are ordered in a bcc-lattice with side $0.2\frac{\pi}{a}$ in the conventional unit cell. The conventional unit cell has the volume $0.2^3(\frac{\pi}{a})^3$ but contains two lattice points (two **q**-vectors). Each **q**-vector will therefore correspond to the volume $\Delta\mathbf{k} = \frac{4}{1000}\left(\frac{\pi}{a}\right)^3$ Some of the vectors in `qvekt` lie on the boundary of the integration volume. Their corresponding unit volume is partly outside the boundary, and this part should not be taken into account when calculating the integral. The last column in this file contains the weight W that the contribution for each vector shall be multiplied by, to take into account that we are only integrating over 1/48 part and that some points lay on the boundary. The final expression becomes:

$$\frac{C_V}{V} = \frac{1}{(2\pi)^3} \frac{4}{1000} \left(\frac{\pi}{a}\right)^3 \sum_j \sum_{\mathbf{q}} W(\mathbf{q}) f_j(\mathbf{q}). \tag{20}$$

Exercise 3:

Calculate the heat capacity (C_V/V) as a function of temperature.

6 Programming specification

Exercise 2 and 3 should be solved by a common program. The name of the program should be:

`phonons`

6.1 Program function

The following sections specify the function of the program. All input (except the `qvekt` file) should be in the form of command line arguments, and all output should be on standard output (`stdout`). Error messages should be printed on `stderr`. All physical quantities should be in the proper SI unit.

Please note that the instructions given here and below must be followed **in every detail** to enable automatic testing of the program!

6.1.1 Phonon frequencies

To calculate phonon frequencies the program should be invoked as:

`phonons` *substance* `omega` q_{x1} q_{y1} q_{z1} [q_{x2} q_{y2} q_{z2} [*npoints*]],

which should calculate the phonon frequencies of the specified substance. The frequencies should be calculated in *npoints* points along a line (in **q**-space) starting at (q_{x1}, q_{y1}, q_{z1}) and ending at (q_{x2}, q_{y2}, q_{z2}) (in equal intervals). If *npoints* is not given eleven points should be used, if also q_{x2}, q_{y2}, q_{z2} is omitted only the single point (q_{x1}, q_{y1}, q_{z1}) should be used. Verify that all parameters are correct, if they are not print a message on `stderr` and exit with a non-zero status. (Note that the square parenthesis in the specification encloses optional arguments).

For each **q** the program should print a single line on standard output:

q_x q_y q_z ω_1 ω_2 ω_3

where the phonon frequencies are printed in ascending order.

6.1.2 γ

To calculate γ the program should be invoked as:

`phonons` *substance* `gamma` q_{x1} q_{y1} q_{z1} [q_{x2} q_{y2} q_{z2} [*npoints*]],

which should calculate the phonon frequencies of the specified substance. The parameters has the same meaning here as when calculating the phonon frequencies. Verify that all parameters are correct, if they are not print a message on `stderr` and exit with a non-zero status.

For each **q** the program should print a single line on standard output:

q_x q_y q_z γ_1 γ_2 γ_3

where the γ values should be printed in the order which corresponds to the order which the frequencies are printed.

6.1.3 Heat capacity

To calculate the heat capacity the program should be invoked as:

`phonons` *substance* `cv` T_1 [T_2 [*npoints*]],

which should calculate C_V/V for the specified substance. C_V/V should be calculated at *npoints* temperatures starting at T_1 and ending at T_2 (in equal intervals). If *npoints* is not given eleven points should be used, if also T_2 is not given C_V/V should only be calculated in the single point T_1. Verify that all parameters are correct, if they are not print a message on `stderr` and exit with a non-zero status.

For each different temperature a single line should be printed on standard output:

T C_V/V

6.1.4 Examples

Here are some example command lines, and what they should do:

```
phonons Ar omega 0.0 0.0 0.0 1.0 0.0 0.0
```

calculate the phonon frequencies of Argon starting at $(0, 0, 0)$ and ending $(1, 0, 0)$ in eleven points.

```
phonons Kr omega 0.0 0.0 0.0 1.0 0.0 0.0 1
```

print an error message an exit with a non-zero status.

```
phonons Kr gamma 0.0 0.0 0.0 1.0 0.0 0.0 13
```

calculate the γ of Krypton starting at $(0, 0, 0)$ and ending $(1, 0, 0)$ in thirteen points.

```
phonons Cu gamma 0.0 0.0 0.0 1.0 0.0 0.0
```

print an error message an exit with a non-zero status.

```
phonons Ar omega 0.0 0.0 0.0 1.0 0.0
```

print an error message an exit with a non-zero status.

```
phonons Ar cv 0.1 100.0 32
```

calculate C_V in 32 points from $T = 0.1\,\mathrm{K}$ to $T = 100\,\mathrm{K}$.

6.2 Program structure

The source code of the programs should be contained in two files:

```
phonons.c
frequencies.c
```

The file `frequencies.c` should contain a routine `frequencies` that calculates the phonon frequencies. The rest of the program (including the `main` routine) should be in the `phonons.c` file. It is wise to split the code in the `phonons.c` file into several functions and/or procedures.

6.2.1 The `frequencies` routine

The `frequencies` routine is specified as:

```
 void frequencies(double A, double B, double m, double *q,
                  double *omega, double *eps)
```

where `A,` `B` and `m` are input parameters specifying the values of A, B and m in the dynamical matrix, eq. (15). The input parameter `q` is a three long array which specifies the **q**-vector. `omega` and `eps` are output parameters, returning the phonon frequencies and the polarisation (eigenvectors) respectively. If **q** is such that one of the eigenvalues of $\hat{D}(\mathbf{k})$ is negative, the corresponding value returned should be $\omega = 0$. `omega` is a three long array, which contains the phonon frequencies in ascending order. `eps` is a nine long array which contains the polarisations in the order

$$\varepsilon_{x1}\ \varepsilon_{y1}\ \varepsilon_{z1}\ \varepsilon_{x2}\ \varepsilon_{y2}\ \varepsilon_{z2}\ \varepsilon_{x3}\ \varepsilon_{y3}\ \varepsilon_{z3}$$

It should be valid to send `eps` as a `NULL`-pointer, in which case the polarisations need not be calculated.

Note that the GSL routine `gsl_eigen_symmv` returns the eigenvalues in no particular order.

6.3 Compiling and linking

A `Makefile` should specify how to compile and link the `phonons` program. The command

```
> make
```

should compile and link.

6.4 Verification

To verify that the specification is fullfiled run the command:

```
> ~fnm/bin/phonons_check
```

in the directory that contains the `phonons` program. While executing various diagnostics are printed (which should be self-explanatory). If the final line of output is:

```
ALL TESTS PASSED
```

then your program fullfills the specification. Otherwise the diagnostics will give an indication of what doesn't work.

If your code doesn't pass the test when you think it should, please contact the supervisor promptly!

6.5 Miscellaneous

6.5.1 Calculating eigenvalues and eigenvectors

A suitable GSL function to calculate the eigenvalues and eigenvectors is `gsl_eigen_symmv`, see the *GSL Reference Manual* for details of how it works. It is declared in the header file `<gsl/gsl_eigen.h>`. Note that workspace must first be allocated with the function `gsl_eigen_symmv_alloc` and that the dynamical matrix must be of type `gsl_matrix`, while the eigenvalues and eigenvectors are returned in a `gsl_vector` and a `gsl_matrix`, respectively.

6.6 Physical constants

To ensure proper precision in the output values, the fundamental physical constants defined in GSL should be used. Alternatively, use the CODATA values, see, *e.g.*, `<http://physics.nist.gov/cuu/Constants/index.html>`.

6.6.1 Numerical command line parameters

Numerical command line parameters can be converted from strings to numbers using either `atof` (for floating point parameters) or `atoi` (for integer parameters). Use of these functions requires `<stdlib.h>`. Here is a small example program that prints the sum of its (floating point) arguments:

```
#include <stdlib.h>
#include <stdio.h>

int
main (int argc, char *argv[])
{

  double sum = 0.0;

  for (int i = 1; i < argc; ++i)
    {
      sum += atof (argv[i]);
    }

  printf ("The sum of the %d numbers is %.8g\n", argc - 1, sum);

  return 0;

}
```

Presentation of results

Present your results in an extensive report. Your report should be written as a project report without reference to the instructions and should among other things include

- the solution to exercise 1;
- comments on the result of excercise 3 (including a discussion of how one can calculate the high-temperature limit analytically);
- plots showing the results from excercise 2;
- plots showing the result from exercise 3.

However, your results should be presented in a coherent text. **Do not** structure your report in terms of “Exercise 1”, “Exercise 2” and so on. When you write, think of the different exercises as steps on the way of the main project: determining the heat capacity for rare gases. Do not supply program listings, just a note on the side with the path to your source code. **Please note:** the formatting and content description given in the sample report should be followed. Remember that this is a graded report and **your mark will depend on following the instructions properly**.

Please make sure that your program passes the verification test (see section 6.4) *before* you hand in your report. Check with the supervisor to insure that this is the case.

References

[1] N. W. Ashcroft and N. D. Mermin, *Solid State Physics*, 3rd ed. (Saunders, New York, 1976).