

**The following people have participated in creating these solutions:  
Nicolaas E. Groeneboom, Magnus Pedersen Lohne, Karl R. Leikanger**

*NOTE: There might be errors in the solution. If you find something which  
doesn't look right, please let me know*

## Partial solutions to problems: Part 1D

### Problem 1D.1

1. From the lecture notes, Planck's Radiation law is expressed as

$$B(\nu) = \frac{2h\nu^3}{c^2} \frac{1}{e^{h\nu/(kT)} - 1}$$

At large frequencies (or high energies),  $h\nu \gg kT$ . Then  $e^{h\nu/(kT)} - 1 \sim e^{h\nu/(kT)}$ , as the exponential dominates. In this limit,

$$B(\nu) = \frac{2h\nu^3}{c^2} e^{-h\nu/(kT)}$$

2. When frequencies are very low, then  $h\nu \ll kT$ . In this limit, we use that for small values of  $x$ , then  $e^x \sim 1 + x$  (Taylor expansion) such that

$$B(\nu) = \frac{2h\nu^3}{c^2} \frac{1}{e^{h\nu/(kT)} - 1} \sim \frac{2h\nu^3}{c^2} \frac{kT}{1 + h\nu - 1} = \frac{2kT}{c^2} \nu^2$$

3. Small frequencies correspond to large wavelength, so I radio astronomers are most used to this equation.

### Problem 1D.2

1. Do it yourself
2. Do it yourself
3. You can start with the following code:

```
from scitools.all import *

#Function that calculates the value of eq. (8) for all elements in the x-array
def function():
    f = ...
    plot_function(f, filename)    #Call plot function
    return f

#Plot-function
def plot_function(y, filename):
    ...
```

```

#Function that calculates the difference between the function value and 5
#squared for all values in the x-array
def diff():
    delta = ... #Error-array
    plot_function(delta, filename) #Call plot function
    return delta

#Function that finds the x-value which gives the minimum value of
#the delta-function
def best_value():
    ... #Declare best-variables
    for i in (...): #For-loop. Go through every element in the delta-array
        if delta[i] < best_delta: #Check if you have a better x-value
            ... #Update best-variables
    print '...' #Write best x-value to screen

#----
#MAIN

#Constants
...

#Make x-array
x = linspace(x_min, x_max, 'number of points')

#Calculations
func = function()
delta = diff()
best_value()

```

### Problem 1D.3

You can find the answer to all parts of this exercise by reading the lecture notes carefully.

### Problem 1D.4

1. We will use the Sun - Earth system here. Total incoming power on earth's surface:

$$P_{\text{earth}}^{\text{in}} = F_{\text{earth}}^{\text{in}} \pi r_{\text{earth}}^2 \approx 1367W \cdot \pi \cdot (6371km)^2 \approx 1.74 \cdot 10^{17}W$$

2. Estimate of earth's temperature:

$$T_{\text{earth}} = \left( \frac{F_{\text{earth}}^{\text{out}}}{\sigma} \right)^{1/4} = \left( \frac{L_{\text{earth}}^{\text{out}} / (4\pi r_{\text{earth}}^2)}{\sigma} \right)^{1/4} \approx \left( \frac{1.74 \cdot 10^{17} \text{W} / (4\pi (6371 \text{km})^2)}{5.67 \cdot 10^{-8} \text{W/m}^2 \text{K}^4} \right)^{1/4} \approx 279 \text{K}$$

Since we consider the earth as a blackbody, the total emitted power is exactly the same as the total incoming and absorbed power ( $F_{\text{earth}}^{\text{out}} = F_{\text{earth}}^{\text{in}}$ ). A blackbody emits the same amount in all directions, so the outgoing fluxdensity must be smaller than the incoming ( $F_{\text{earth}}^{\text{in}} > F_{\text{earth}}^{\text{out}}$ ).

## Problem 1D.5

1. The intensity is  $B(\lambda)$  is defined as

$$B(\lambda) = \frac{dE}{dAd\Omega dt d\lambda \cos \theta}$$

such that

$$\int B(\lambda) \cos \theta d\Omega dA = \frac{dE}{dt d\lambda} \equiv L(\lambda)$$

$B(\lambda)$  is regarded as a constant when integrating  $dAd\Omega$ . Integrating the solid area ( $d\Omega = \sin \theta d\theta d\phi$ ), we find

$$\int_{\text{half the sphere}} \cos \theta d\Omega = \int_0^{2\pi} \int_0^{\pi/2} \cos \theta \sin \theta d\theta d\phi$$

The  $\phi$  part is constant, such that

$$= 2\pi \cdot \int_0^{\pi/2} \cos \theta \sin \theta d\theta = 2\pi \cdot \frac{1}{2} = \pi$$

Integration of the area of the sphere gives  $A = 4\pi R^2$ , where  $R$  is the radius of the star. Hence

$$L(\lambda) = 4\pi^2 R^2 B(\lambda)$$

2. At distance  $r$  from the star, the area of the sphere is given as  $4\pi r^2$ . As the flux = luminosity / area, we obtain

$$F(\lambda) = L(\lambda) / \text{area} = \frac{4\pi^2 R^2}{4\pi r^2} B(\lambda) = \pi \frac{R^2}{r^2} B(\lambda)$$

3. When obtaining Wien's displacement law,  $B(\lambda)$  was differentiated and set equal to 0. This could also have been done using  $F(\lambda)$ , as  $F(\lambda)$  only depends on  $\lambda$  through  $B(\lambda)$ . *Morale: To determine the wavelength of maximum intensity (and thereby the colour temperature) of an object, using the measured **flux** directly instead of intensity works just as fine.*

## Problem 1D.6

This is a suggestion to a code which could be used:

```
from scitools.all import *

#DAY-CLASS
class day:
    #Constructor
    def __init__(self, day):
        #Declare self-variables/lists
        self.day = day
        self.filename = day + '.txt'
        self.lambda_obs = []
        self.flux = []
        ...

        #Calculations/Call functions
        self.read_data()
        self.plot_spec()
        self.star_velocity()
        self.model()

    #Function that reads data from self.filename
    def read_data(self):
        ...
        ...           #Store data in self-lists
        ...           #Convert from lists to arrays

    #Function that plots the flux vs observed wavelength
    def plot_spec(self):
        ...

    #Function that calculates the velocity of the star
    def star_velocity(self):
        #Read in (by eye measurement) the wavelength of minimum flux
        lambda_min = float(raw_input('...'))

        #Calculate radial velocity by Doppler's formula
        v_rad = ...

        #Write result to file - Same file for all days
        if self.day == 'day0':
            file = open(filename, 'w')
        else:
            file = open(filename, 'a')
```

```

...          #Write to file
file.close()

#Function that calculates the best value of the parameters Fmin, sigma
#and lambda_central, and computes the radial velocity of the star by using the
#model-value of lambda_central
def model(self):
    #Read in min and max values of Fmin, sigma and lambda_central
    F_min = float(raw_input('...'))
    F_max = float(raw_input('...'))
    ...
    ...

    #Declare arrays
    points = ...          #Number of evaluation-points per parameter
    Fmin = linspace(F_min, F_max, points) #Fmin array
    sigma = ...          #sigma array
    lambda_central = ...          #lambda_central array

    #Declare best-variables
    best_Fmin = ...
    ...

    #Determine the best values of Fmin, sigma and lambda_central by
    #least-square method
    for i in range(points):
        for j in range(points):
            for k in range(points):
                flux_model = ...          #Calculate the model-value by eq. (9)
                delta = sum((...)**2)    #Calculate the error-array
                if delta < best_delta:   #Check if you got a better const. set
                    ...                  #Update best-variables

    #Write the best parameters to file - Collect the parameters for each day
    #in the same file (see pervious problem)
    ...

    #Calculate the radial velocity of the star by using the best value of
    #lambda_central
    ...

    #Write the radial velocity to file - Collect the values for each day in
    #the same file
    ...

```

## Problem 1D.8

We use that the absolute magnitude is given as

$$M = m - 5\log_{10}\left(\frac{r}{10pc}\right)$$

### Sirius

Sirius resides at a distance of 8.6ly with apparent magnitude  $m = -1.47$  such that  $M \sim 1.45$ .

### Vega

Vega resides at a distance of 24.27ly with apparent magnitude  $m = 0.0$  such that  $M \sim 0.56$ .

### Sun

The Sun resides at a distance of 1AU with apparent magnitude  $m = -26.7$  such that  $M \sim 4.87$ .

## Problem 1D.9

1. The flux of the Sun at Earth-distance was in a previous exercise found to be  $F_{sun} = 1367J/s/m^2$ . The luminosity of Vega is (wiki)  $L_{vega} = 37L_{sun}$ , such that the recieved flux at earths distance from vega ( $r_v = 24.27ly$ ) is given as

$$F_{vega} = \frac{L_{vega}}{4\pi r_v^2} = \frac{37L_{sun}}{4\pi(24.27ly)^2} \approx 2.7 \cdot 10^{-8} J/s/m^2.$$

Using that the apparent magnitude of Vega is zero, we have

$$m_1 - m_2 = m_1 = -2.5\log_{10}\left(\frac{1337}{2.7 \cdot 10^{-8}}\right) \approx -26.7$$

2. Let  $m_1 = -26.7$  be the apparent magnitude of the Sun, while  $m_2 = 30$  is the maximum apparent magnitude. Using

$$m_1 - m_2 = -2.5\log_{10}(F_1/F_2) = -2.5\log_{10}\frac{L/(4\pi(1AU)^2)}{L/(4\pi r^2)} = -5\log_{10}(1AU/r)$$

such that

$$r = 1AU \cdot 10^{\frac{30+26.7}{5}} \approx 1.1Mpc$$

3. This problem is solved as in the previous question, but where the luminosities cancelled, we're now left with  $2 \cdot 10^{11}$ . This implies:

$$r = 1AU \cdot \sqrt{(2 \cdot 10^{11})} \cdot 10^{\frac{m_2 - m_1}{5}} \approx 447Gpc$$

This is well beyond the observable universe, but looking at large distances also means looking backwards in time to an epoch when no galaxies existed. We will therefore never see galaxies at very large distances.