Plot functions

- Plot a function with single variable
- Syntax: **function[x_]:=**; **Plot[f[x],{{x,xinit,xlast}}]**; **Plot[{f1[x],f2[x]},{x,xinit,xlast}]**;
- Plot various single-variable functions in Chapter 1, ZCA 110 as examples.
- Plot a few functions on the same graph.

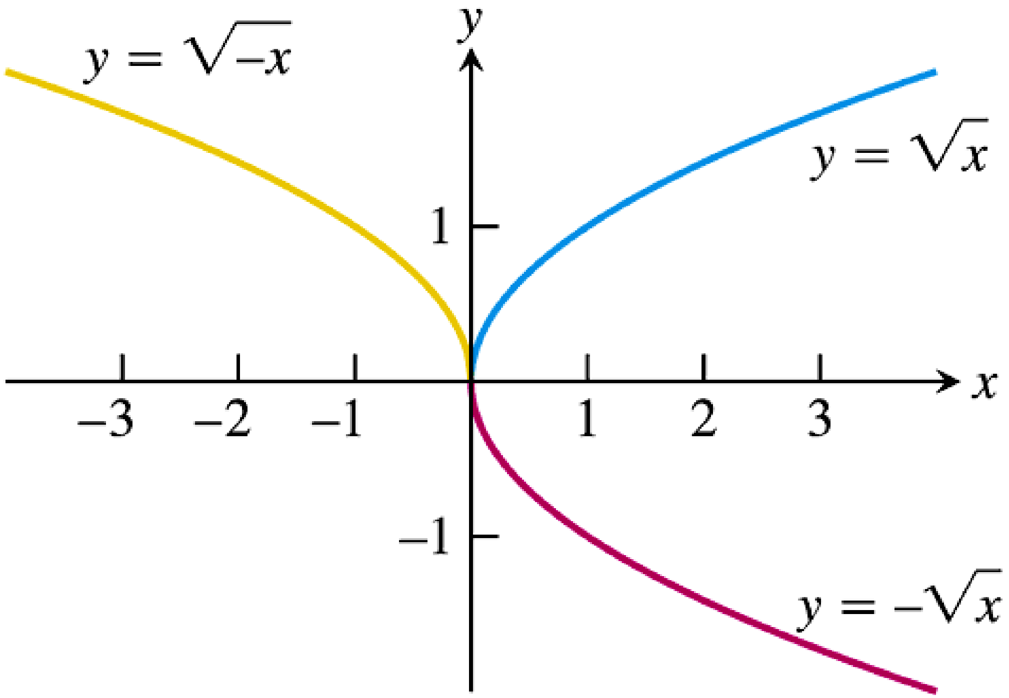


FIGURE 1.59 Reflections of the graph $y = \sqrt{x}$ across the coordinate axes (Example 5c).

Plot a few functions on the same graph

- **F1[x_]:=1*x;**
- **F2[x_]:=2*x;**
- **F3[x_]:=3*x;**
- **F4[x_]:=4*x;**
- **list={F1[x],F2[x],F3[x],F4[x]};**
- **Plot[list,{x,-10,10}]**

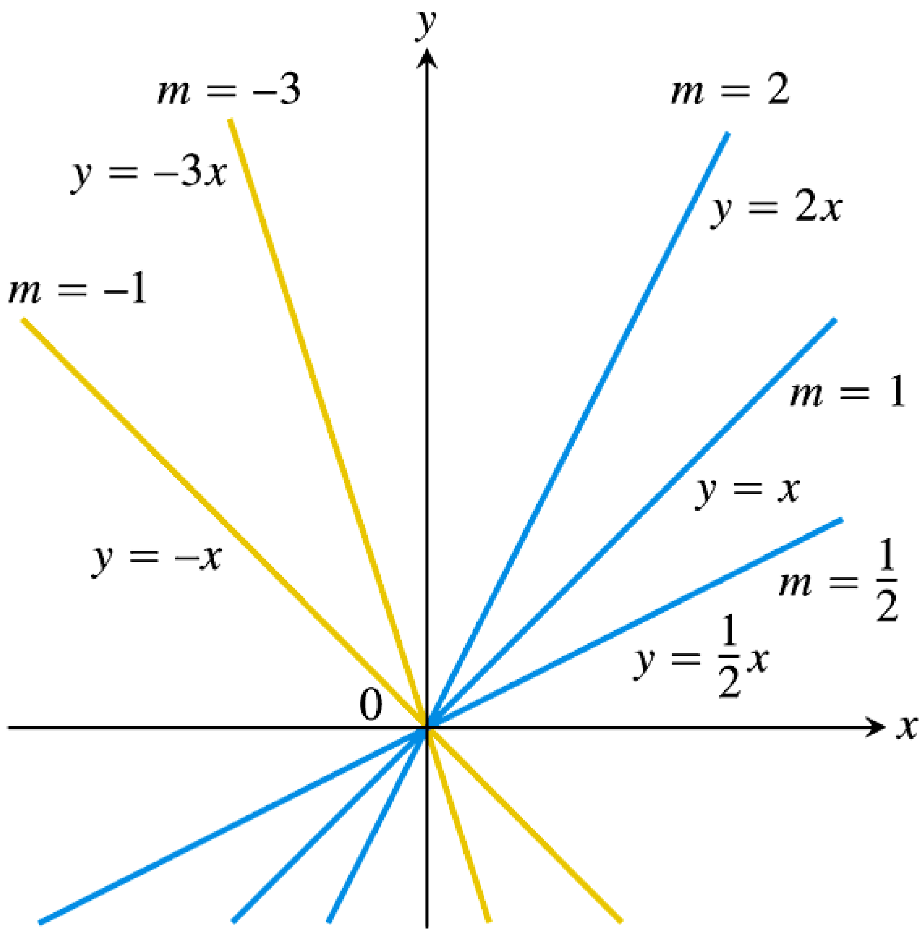


FIGURE 1.34 The collection of lines $y = mx$ has slope m and all lines pass through the origin.

Plot a few functions on the same graphs

- Do the same thing by defining the functions to depend on x and n:
- **F[x_,n_]:=n*x;**
- **list={F[x,1], F[x,2], F[x,3], F[x,4]};**
- **Plot[list,{x,-10,10}]**

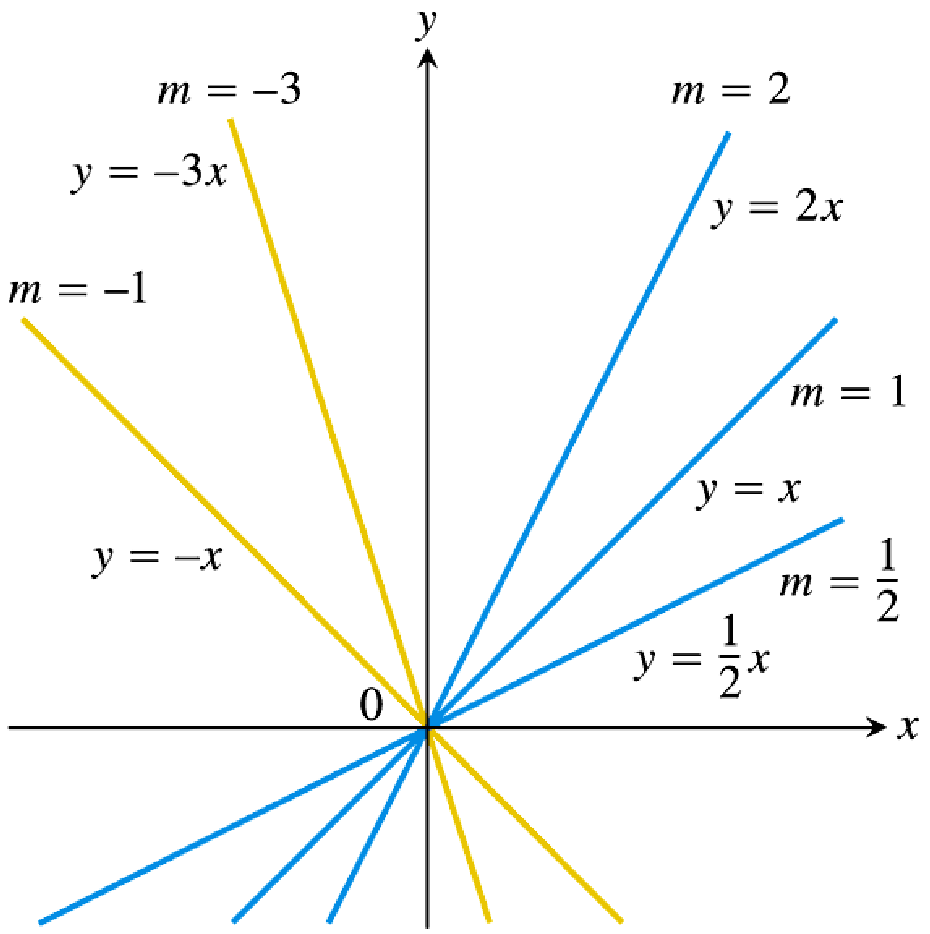
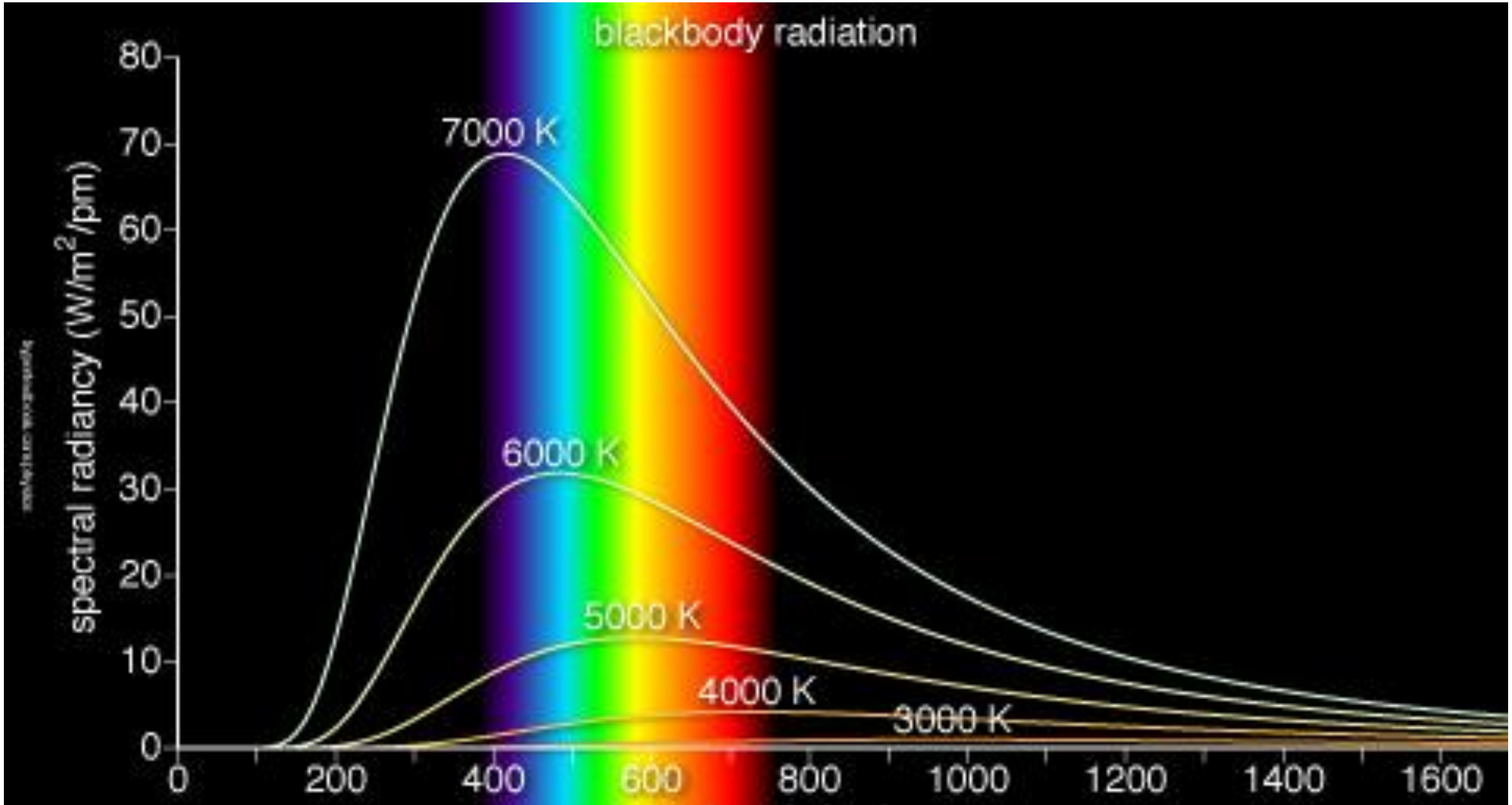


FIGURE 1.34 The collection of lines $y = mx$ has slope m and all lines pass through the origin.

Black Body Radiation

$$R(\lambda,T)=\frac{2\pi hc^2}{\lambda^5\left(e^{hc/\lambda kT}-1\right)}$$



Exercise

- Plot Planck's law of black body radiation for various temperatures on the same graph by defining R as a function of two variables.
- Define function of two variables: $R(\lambda,T)=\dfrac{2\pi hc^2}{\lambda^5\left(e^{hc/\lambda kT}-1\right)}$
- h,c,T, are constants;
- **R[lambda_,T_]:=2Pi*h*c^2/(lambda^5*(Exp[h*c/(lambda*k*T)]-1));**
- Customize the plots using these:
- **PlotLabel**; **AxesLabel**; **PlotLegend;PlotRange;**

Exercise:

- Manually locate λ_{max}, the wavelength at which *R*(λ,*T*) is maximum for a fixed *T*.
- Write a Do loop to automatically do this.
- Hence, generate the list
- {{λ_{max1},T1}, {λ_{max2},T2}, {λ_{max3},T3}, {λ_{max4},T4},…}

Hence, proof Weinmann’s displacement law.

Syntax: Table[]; Sum[]

- Generate a list using **Table[f[x,n], {n,ninit,nlast}]**;
- The function $f(x, N_0) = \sum_{n=1}^{n=N_0} x^n$ can be expressed in Mathematica as
- **F[x_,N0_]:=Sum[x^n,{n,1,N0}];**
- Use these to numerically verify that the infinite series representation of a function converges into the function.

EXAMPLE 4 Applying Term-by-Term Differentiation

Find series for $f'(x)$ and $f''(x)$ if

$$f(x) = \frac{1}{1-x} = 1 + x + x^2 + x^3 + x^4 + \cdots + x^n + \cdots$$
$$= \sum_{n=0}^{\infty} x^n, \qquad -1 < x < 1$$

EXAMPLE 6 A Series for $\ln(1+x)$, $-1 < x \leq 1$

The series

$$\frac{1}{1+t} = 1 - t + t^2 - t^3 + \cdots$$

converges on the open interval $-1 < t < 1$.

$$\ln(1+x) = \int_0^x \frac{1}{1+t}\,dt = t - \frac{t^2}{2} + \frac{t^3}{3} - \frac{t^4}{4} + \cdots \Big]_0^x$$
$$= x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots, \qquad -1 < x < 1.$$

Example 2 Finding Taylor polynomial for e^x at $x = 0$

$$f(x) = e^x \to f^{(n)}(x) = e^x$$

$$P_n(x) = \sum_{k=0}^{k=n} \frac{f^{(k)}(x)}{k!}\bigg|_{x=0} x^k = \frac{e^0}{0!}x^0 + \frac{e^0}{1!}x^1 + \frac{e^0}{2!}x^2 + \frac{e^0}{3!}x^3 + \ldots \frac{e^0}{n!}x^n$$

$$= 1 + x + \frac{x^2}{2} + \frac{x^3}{3!} + \ldots \frac{x^n}{n!}$$ This is the Taylor polynomial of order n for e^x

If the limit $n \to \infty$ is taken, $P_n(x) \to$ Taylor series.

The Taylor series for e^x is $1 + x + \frac{x^2}{2} + \frac{x^3}{3!} + \ldots \frac{x^n}{n!} + \ldots = \sum_{n=0}^{\infty} \frac{x^n}{n!}$,

In this special case, the Taylor series for e^x converges to e^x for all x.

Exercise: Numerical verification of $\langle \varepsilon \rangle = \frac{h\nu}{e^{h\nu/kT}-1}$

$$\langle \varepsilon \rangle = \frac{\sum_{n=0}^{\infty} N(n) E_n}{\sum_{n=0}^{\infty} N(n)}; N(n) = N_0 nh\nu; E_n = nhf; n = 0,1,2,3,\ldots$$

$$\langle \varepsilon \rangle = \frac{\sum_{n=0}^{\infty} N_0 nh\nu \exp\left(-\frac{nh\nu}{kT}\right)}{\sum_{n=0}^{\infty} N_0 \exp\left(-\frac{nh\nu}{kT}\right)}$$

$$= \frac{0 + h\nu e^{-\frac{h\nu}{kT}} + 2h\nu e^{-\frac{2h\nu}{kT}} + 3h\nu e^{-\frac{3h\nu}{kT}} + \cdots}{1 + e^{-\frac{h\nu}{kT}} + e^{-\frac{2h\nu}{kT}} + e^{-\frac{3h\nu}{kT}} + \cdots}$$

$$= \cdots = \frac{h\nu}{e^{h\nu/kT}-1}$$

- Expectation value of a photon's energy when deriving Planck's law for black body radiation;
- Define $x = h\nu/kT$
- The sum over all *n* in the RHS

$$\frac{\sum_{n=0}^{\infty} N_0 nh\nu \exp(-nx)}{\sum_{n=0}^{\infty} N_0 \exp(-nx)}$$

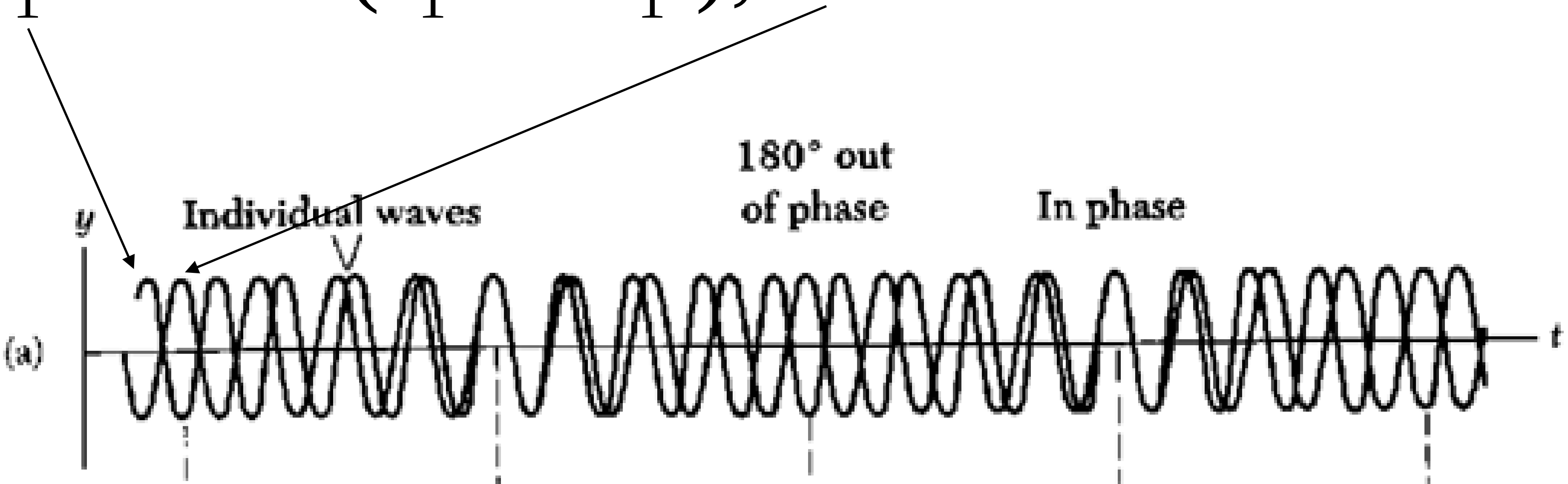
should converge to $\langle \varepsilon \rangle = \frac{h\nu}{e^{h\nu/kT}-1}$

in the limit *n* → infinity.

Constructing wave pulse

- Two pure waves with slight difference in frequency and wave number $\Delta\omega = \omega_1 - \omega_2$, $\Delta k = k_1 - k_2$, are superimposed

$$y_1 = A\cos(k_1 x - \omega_1 t); \quad y_2 = A\cos(k_2 x - \omega_2 t)$$

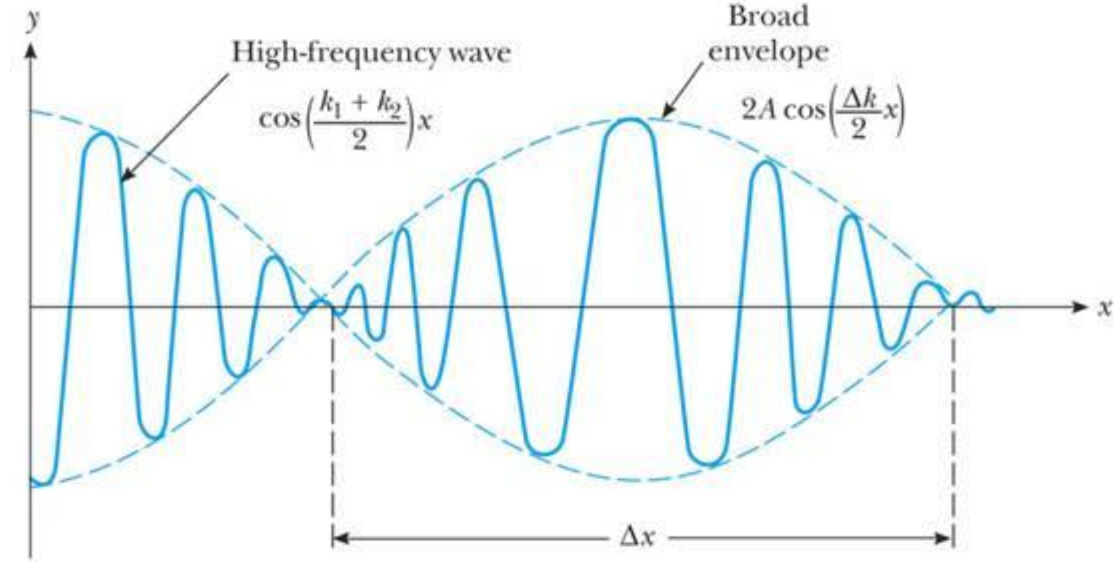


Envelop wave and phase wave

The resultant wave is a 'wave group' comprise of an `envelop' (or the group wave) and a phase waves

$$y = y_1 + y_2$$

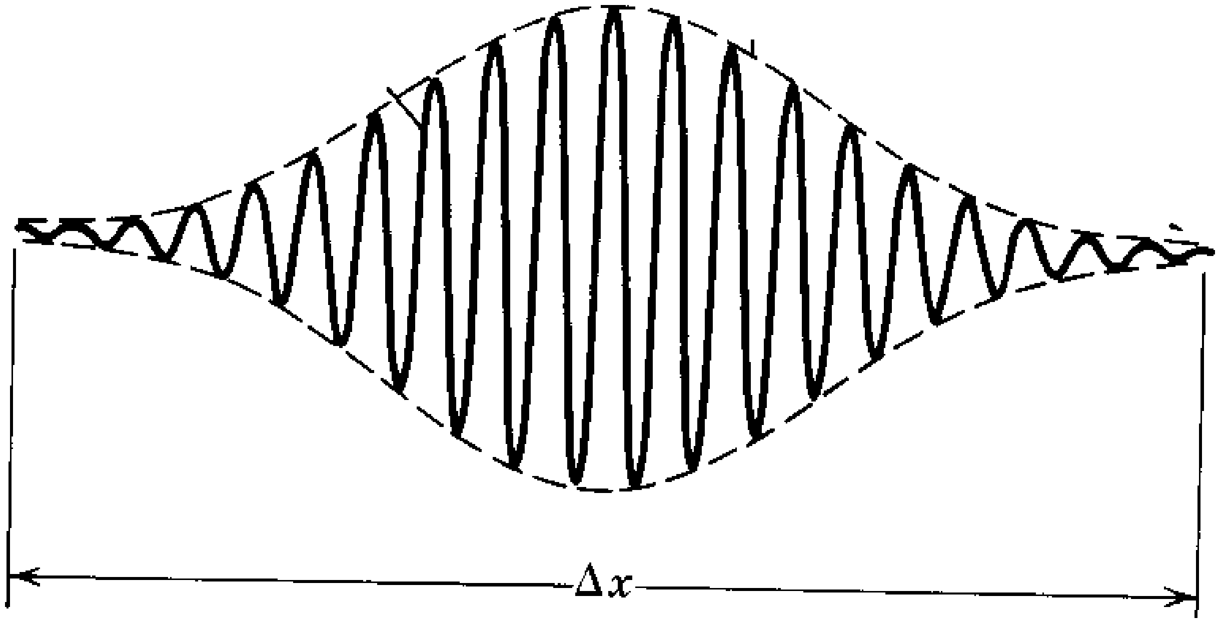
$$= 2A\cos\frac{1}{2}\left(\{k_2 - k_1\}x - \{\omega_2 - \omega_1\}t\right)\cdot\cos\left\{\left(\frac{k_2 + k_1}{2}\right)x - \left(\frac{\omega_2 + \omega_1}{2}\right)t\right\}$$



Wave pulse – an even more \`localised' wave

- In the previous example, we add up only two slightly different wave to form a train of wave group
- An even more \`localised' group wave – what we call a "*wavepulse*" can be constructed by adding more sine waves of different numbers k_i *and* possibly different amplitudes so that they interfere constructively over a small region Δx and outside this region they interfere destructively so that the resultant field approach zero
- Mathematically,

$$y_{\text{wave pulse}} = \sum_{i}^{\infty} A_i \cos\left(k_i x - \omega_i t\right)$$



A wavepulse – the wave is well localised within Δx. This is done by adding a lot of waves with with their wave parameters $\{A_i, k_i, \omega_i\}$ slightly differ from each other (i = 1, 2, 3….as many as it can)

Exercise: Simulating wave group and wave pulse

- Construct a code to add *n* waves, each with an angular frequency omegai and wave number ki into a wave pulse for a fixed t.
- Display the wave pulse for t=t0, t=t1, …, t=tn.
- Syntax: **Manipulate**
- Sample code: wavepulse.nb

Syntax: ParametricPlot[], Show[]

- The trajectory of a 2D projectile with initial location (x_0, y_0), speed v_0 and launching angle θ are given by the equations:
- $x(t) = x_0 + v_0 t \cos\theta; y(t) = y_0 + v_0 t \sin\theta + \frac{g}{2} t^2$, for t from 0 till *T*, defined as the time of flight, *T* =-$2(y_0 + v_0 \sin\theta)/g$.
- $g = -9.81;$
- The trajectories can be plotted using **ParametricPlot**.
- You can combine few plots using **Show[]** command.

2D projectile motion (recall your Mechanics class)

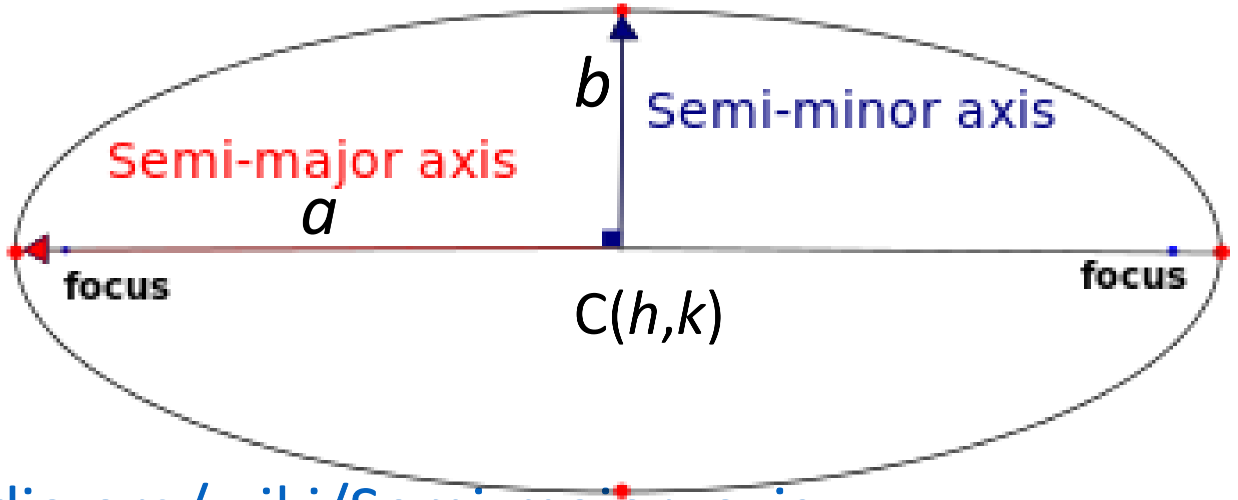
- Plot the trajectories of a 2D projectile launched with a common initial speed but at different angles
- Plot the trajectories of a 2D projectile launched with a common angle but different initial speed.
- Sample code: 2Dprojectile.nb
- For a fixed v0 and theta, how would you determine the maximum height numerically (not using formula)?

Exercise: Circular motion

- Write down the parametric equations for the x and y coordinates of an object executing circular motion.

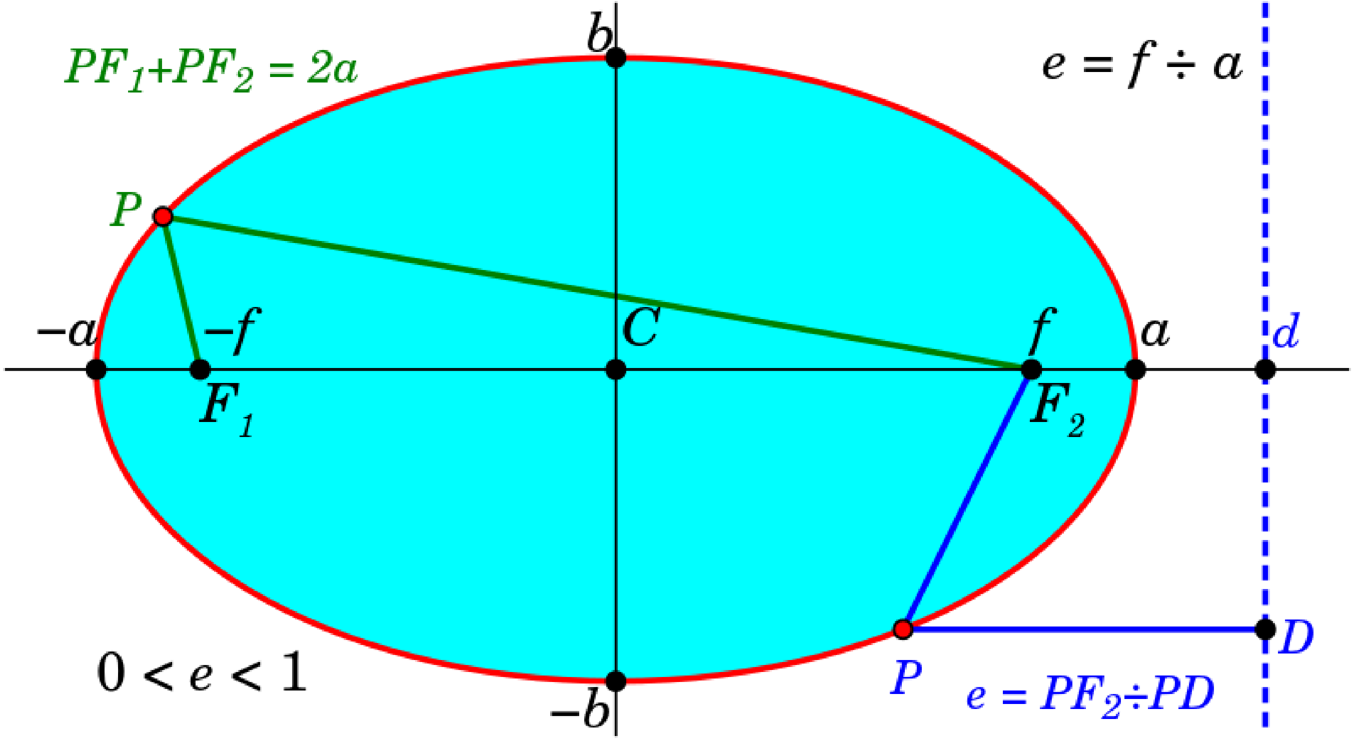
- Plot the trajectories of a particle moving in a circle (recall your vector analysis class, ZCT 211)

Parametric Equation of an Ellipse



- http://en.wikipedia.org/wiki/Semi-major_axis
- In geometry, the **major axis** of an ellipse is its longest diameter: line segment that runs through the center and both foci, with ends at the widest points of the perimeter. The **semi-major axis, *a*,** is one half of the major axis, and thus runs from the centre, through a focus, and to the perimeter. Essentially, it is the radius of an orbit at the orbit's two most distant points. For the special case of a circle, the semi-major axis is the radius. One can think of the semi-major axis as an ellipse's *long radius*.

Geometry of an ellipse

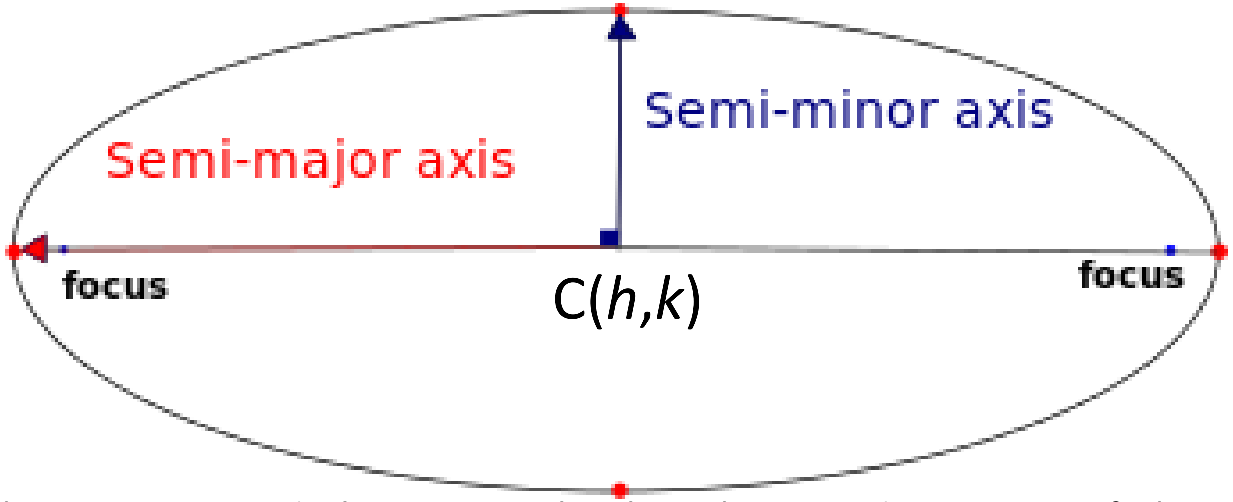


The distance to the focal point from the center of the ellipse is sometimes called the **linear eccentricity**, *f*, of the ellipse.

In terms of semi-major and semi-minor, $f^2 = a^2 - b^2$.

e is the eccentricity of an ellipse is the ratio of the distance between the two foci, to the length of the major axis or $e = 2f/2a = f/a$

Elliptic orbit of a planet around the Sun



- Consider a planet orbitng around the Sun which is located at one of the foci of the ellipse.
- Coordinates of the planet at time *t* can be expressed in parametrised form:
- $x(t) = h + a \cos \omega t$; $y = k + b \sin \omega t$; or equivalently, $\left(\frac{x-h}{a}\right)^2 + \left(\frac{y-k}{b}\right)^2 = 1$

where *x*, *y* are the coordinates of any point on the ellipse at time *t, a*, *b* are semi-major and semi-minor.

- (*h*,*k*) are the *x* and *y* coordinates of the ellipse's center.
- ω is the angular speed of the planet. ω is related to the period *T* of the planet via $T=2\pi/\omega$; whereas the period *T* is related to the parameters of the planetary system via $T = 2\pi\sqrt{\frac{a^3}{GM}}$, where *M* is the mass of the Sun.

Exercise: Marking a point on a 2D plane.

- $x = h + a \cos \omega t$; $y = k + b \sin \omega t$. Set $\omega=1$.
- Display the parametric plot for an ellipse with your choice of *h, k, a, b.*
- How would you mark a point with the coordinate (x(t),y(t)) on the ellipse?
- Syntax: **ListPlot[{{x[t],y[t]}}]**;
- You can customize the size of the point using **PlotStyle->PointSize[0.05], PlotMarkers;**

Exercise: Simulating an ellipse trajectory in 2D

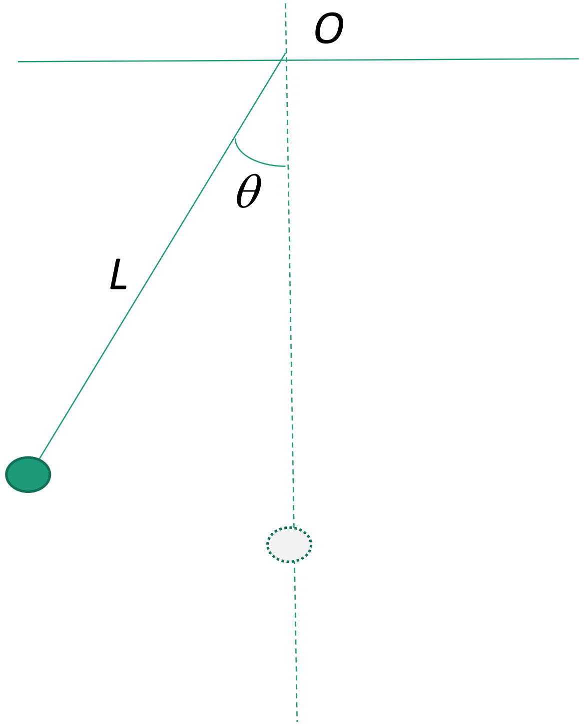
- How would you construct a simulation displaying a point going around the ellipse as time advances?
- Sample code: ellipse1.nb

Exercise:

- (i) Given any moment *t*, how would you abstract the coordinates of a point P(*t*) on the ellipse?
- (ii) How could you obtain the coordinates P'(*t*) at the other end of the straight line connecting to point P(*t*) via the center point (*h*,*k*)? (you have to think!)
- (iii) Given the knowledge of P(*t*) and P'(*t*), draw a line connecting these two points on the ellipse (see sample code 3 in ellipse1.nb) at fixed *t*.
- (iv) Simulate the rotation of the straight line about (*h*,*k*) as the point P move around the ellipse.
- (v) Use your code to "measure" the maximum and minimum distances between the points PP' (known as major axis and minor axis). Theoretically, major axis = Max[2*b*,2*a*]; minor axis = Min[2*b*,2*a*]; see ellipse2.nb

Exercise: Simulating SHM

- A pendulum executing simple harmonic motion (SHM) with length L, released at rest from initial angular displacement θ_0, is described by the following equations: $\theta(t) = \theta_0 \cos \omega t, \omega = \sqrt{\frac{g}{L}}$. The period T of the SHM is given by $T = 2\pi/\omega$.
- Simulate the SHM using **Manipulate[]**
- Hint: you must think properly how to specify the time-varying positions of the pendulum, i.e., ($x(t)$,$y(t)$).



See simulate_pendulum.nb

Exercise: Simulating SHM

- Simulate two SHMs with different lengths L1, L2:
- Plot the phase difference between them as a function of time.