

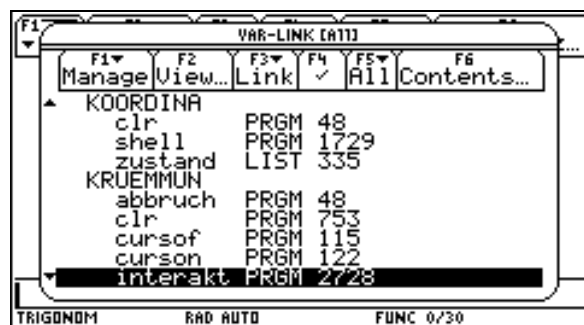
1. Überblick und allgemeine Benützungshinweise:

Die folgenden Programmen sind für die Oberstufe der AHS (Sekundarstufe 2) für den TI-92 entwickelt worden. Wird immer nur eine der Programmgruppen verwendet, dann ist keine Speichererweiterung und auch kein Plusmodul notwendig.

Es folgt ein Überblick, für welche Schulstufe die jeweilige Programmgruppe entwickelt wurde. Diesem Überblick entnimmt man auch eine Kurzbeschreibung der einzelnen Programmgruppen. Außerdem ist der Inhalt des Folders der jeweiligen Programmgruppe abgebildet, damit man die Vollständigkeit aller Dateien und ihre Größe kontrollieren kann.

5. Klasse (9. Schulstufe):

koordina: Übungsprogramm zum Ablesen von Koordinaten in rechtwinkligen Koordinatensystemen



Der Speicherbedarf aller Dateien zusammen beträgt rund 2,2 kB.

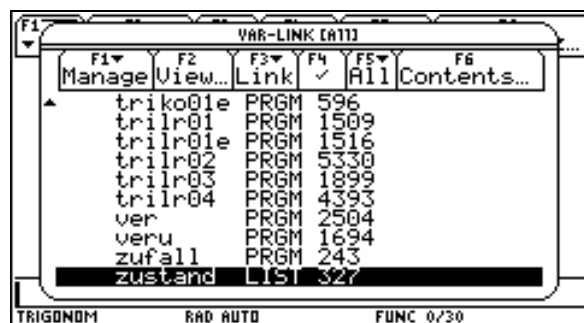
geradgl: Übungsprogramm zum Erkennen von Anstieg und Abschnitt einer linearen Funktion aus ihrem Graphen



Der Speicherbedarf aller Dateien zusammen beträgt rund 2kB.

6. Klasse (10. Schulstufe):

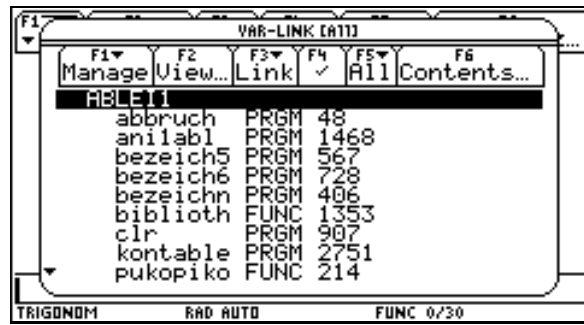
trigonom: Programm zum selbständigen Erlernen der eigenschaften der Funktionen der Form $f(x)=A\sin(wx+p)$



Die Speicherbedarf aller Dateien zusammen beträgt rund 35kB.

7. Klasse (11. Schulstufe):

ablei1: Programm zur interaktiven graphischen Erstellung der 1. Ableitung
 Programm zur Kontrolle der Zusammenhänge zwischen Funktion und 1. Ableitung



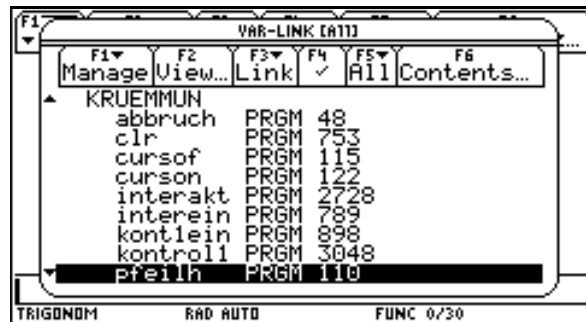
Der Speicherbedarf aller Dateien zusammen beträgt rund 9,7 kB.

able1u2: Programm zur Demonstration der Zusammenhänge zwischen Funktion, 1. und 2. Ableitung



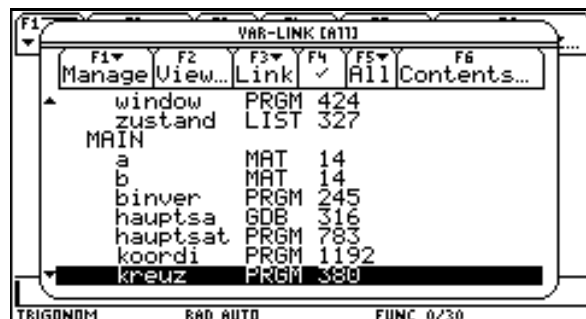
Der Speicherbedarf aller Dateien zusammen beträgt rund 9,3kB.

kruemmun: Programme zum Erlernen und Unterscheiden der Begriffe Funktionswert, Anstieg und Krümmung



Der Speicherbedarf aller Dateien zusammen beträgt rund 9,9kB.

kurvendi: Programme zur Kurvendiskussion von Polynomfunktionen und zum Erkennenlernen der charakteristischen Punkte eines Funktionsgraphen



Der Speicherbedarf aller Dateien zusammen beträgt rund 9,4 kB.

newtonve: Programm zur Demonstration des Newtonschen Näherungsverfahrens



Der Speicherbedarf aller Dateien zusammen beträgt rund 5,7kB.

kegelsch: Programm zur axiometrischen Darstellung der Schnitte eines Drehkegels mit einer Ebene samt Berechnung und Darstellung der entstehenden Kegelschnitte



Der Speicherbedarf aller Dateien zusammen beträgt rund 6,6 kB

8. Klasse (12. Schulstufe):

integral: Interaktive graphische Darstellung des Riemannintegrals mit variabler Obergrenze



Der Speicherbedarf aller Dateien zusammen beträgt rund 6,6 kB.

hauptsat: Programm, um den Hauptsatz der Differential- und Integralrechnung zur verdeutlichen



Der Speicherbedarf aller Dateien zusammen beträgt rund 4,2 kB.

numeriin: Programm zur graphische Darstellung numerischer Integrationsmethoden und ihrer Auswertung





Der Speicherbedarf aller Dateien zusammen beträgt rund 6,4 kB.

rotation: Programm zur axiometrischen Darstellung von Rotationskörpern



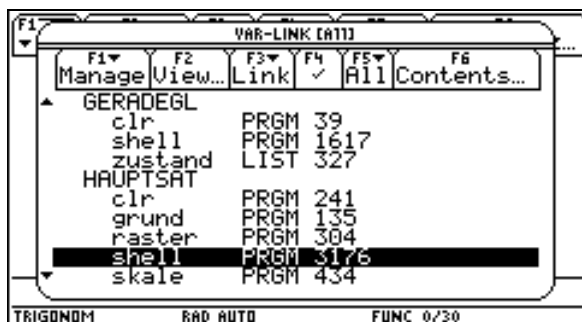
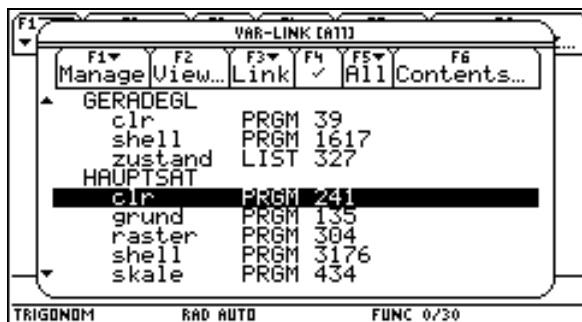
Der Speicherbedarf aller Dateien zusammen beträgt rund 4,4kB.

Allgemeine Benützungshinweise:

Auf der Diskette sind die Programme in doppelter Form abgespeichert. Einerseits sind die Dateien gruppiert vorhanden (z.B. für die Programmgruppe trigonom in der Datei trigonom.92g) andererseits gibt es auch ein Unterverzeichnis (z.B. für die Programmgruppe trigonom das Verzeichnis trigonom), in dem sich die entgrupperten Dateien befinden.

Für alle Programmgruppen gilt der gleiche Startvorgang.

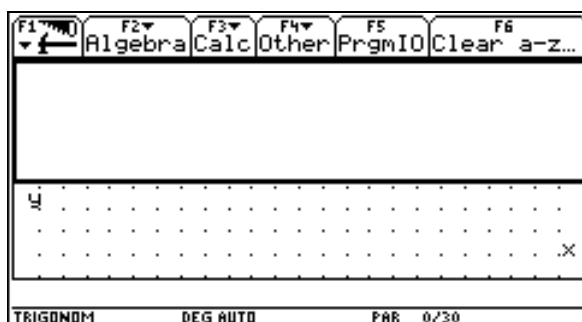
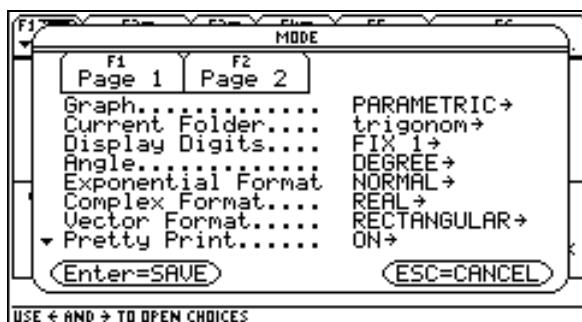
In den entsprechenden Folder wechseln. Also z.B. für die Programmgruppe trigonom in den Folder trigonom wechseln und das Programm **shell()** starten. Das Programm **shell()** ist immer das Startprogramm und startet je nach dem, in welchem Folder man sich befindet, die entsprechende Programmgruppe. Das zweite Program, das in jedem Folder enthalten ist, ist das Programm **clr()**. Es löscht alle von der Programmgruppe angelegten Variablen, damit nie ein Aufräumen im Var-Link nötig ist. Im Normalfall läuft dieses Programm immer beim Beenden einer Programmgruppe ab. Sollte ein Programm doch einmal abgebrochen werden, so kann man das Programm **clr()** starten, um den Folder aufzuräumen



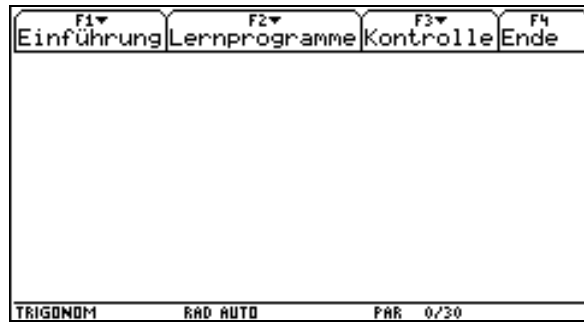
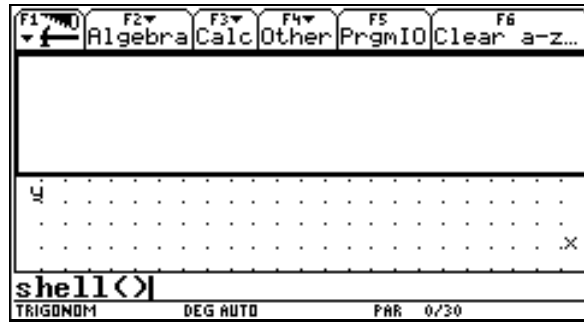
Wenn eine Programmgruppe mit shell() gestartet wird, so werden die MODE-Einstellungen abgespeichert und der Rechner wird nach Beenden des Programmes wieder auf diese Einstellung zurückgeführt.

Im y-Editor abgespeicherte Funktionen und Plots werden durch die Programmgruppen in der Regel gelöscht.

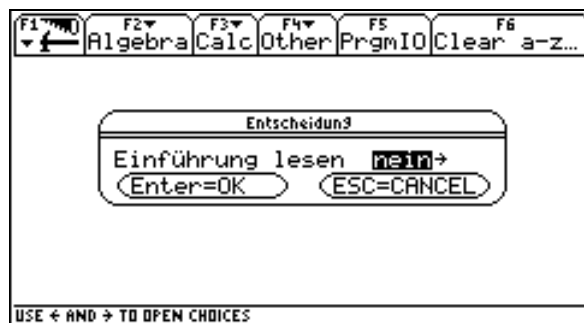
Gehen wir von folgender ausgefallenen Rechneinstellung aus.



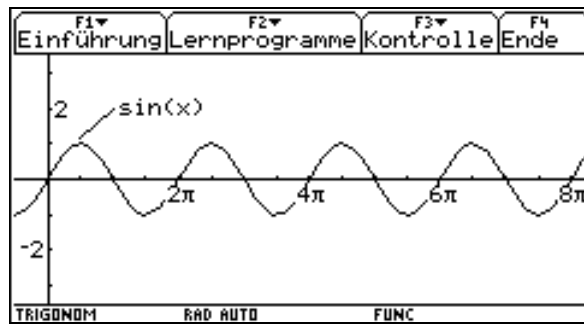
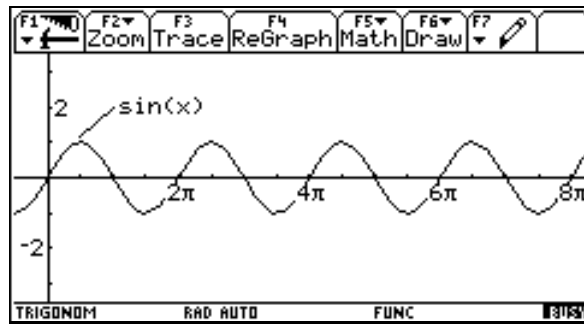
Nun starten wir z.B. shell() (Folder trigonom) und erhalten den Startbildschirm der Programmgruppe trigonom.



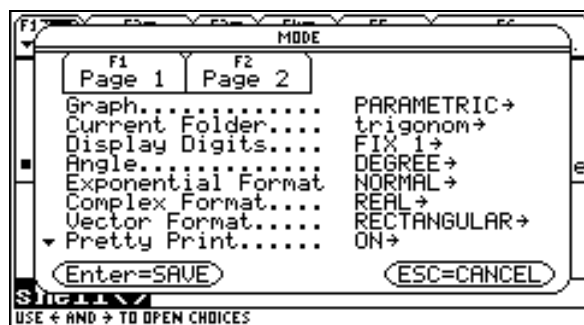
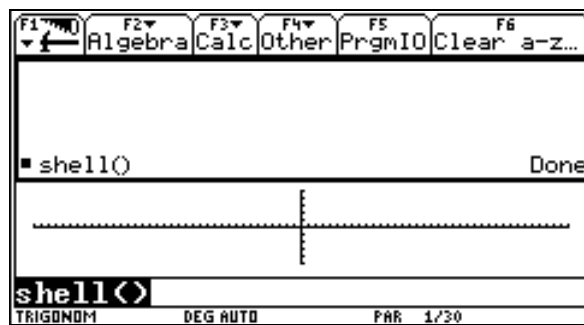
Wir starten probeweise die interaktive Einführung und übergehen das Lesen der Einführung.



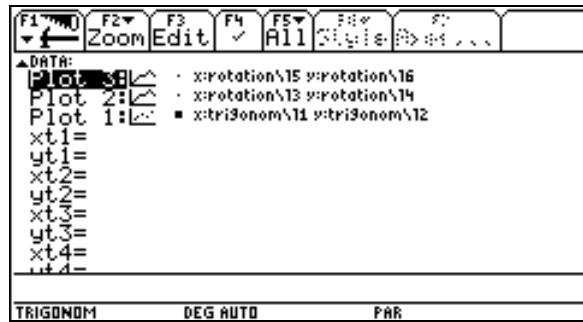
Das kehren wir mit $\text{2nd}[\text{ESC}]$ zur Menüleiste von trigonom zurück und beenden das Programm mit $[\text{F4}]$.



Die alten Modeeinstellungen werden wieder hergestellt..



Der Folder wurde von allen Variablen befreit. Der y-Editor ist gelöscht und eventuell definierte Plots sind deaktiviert.

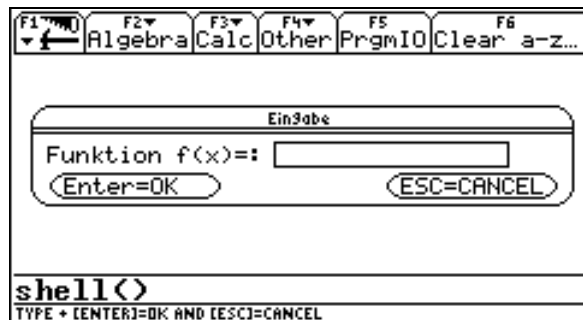


Das Programmverhalten in Dialogboxen ist auch für alle Programmgruppen standardisiert.

[ESC] bedeutet immer, dass das Programm beendet wird.

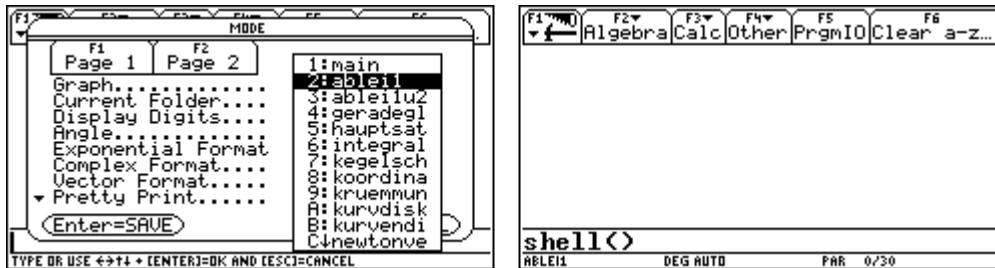
[ENTER] bewirkt ein Fortsetzen mit der gewählten Einstellung.

Ist in der Dialogbox eine Eingabe zu tätigen und wurde sie vergessen, so bewirkt [ENTER], dass die Dialogbox wieder erscheint (in der abgebildeten Box, wenn auf die Eingabe der Funktion vergessen wird). Fehleingaben bewirken immer einen Absturz des Programmes (in der abgebildeten Box eine Funktion in Abhängigkeit von t).

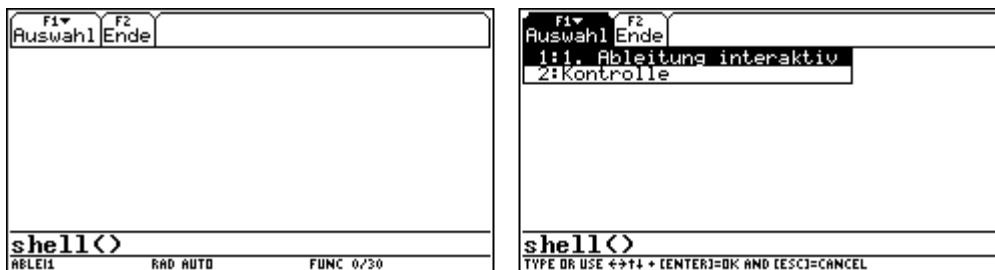


2. Programmgruppe: Ablei1:

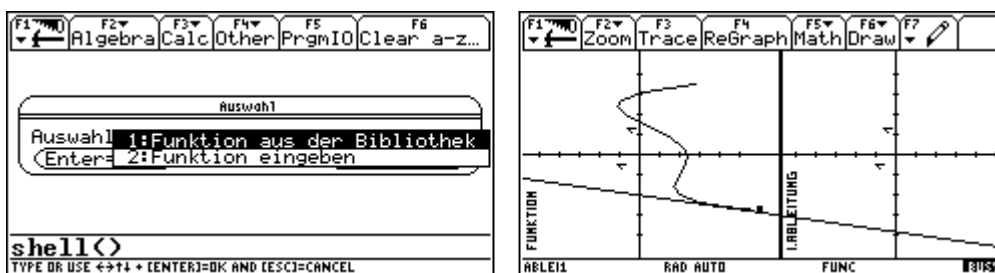
In den Folder Ablei1 wechseln und das Programm shell() starten.



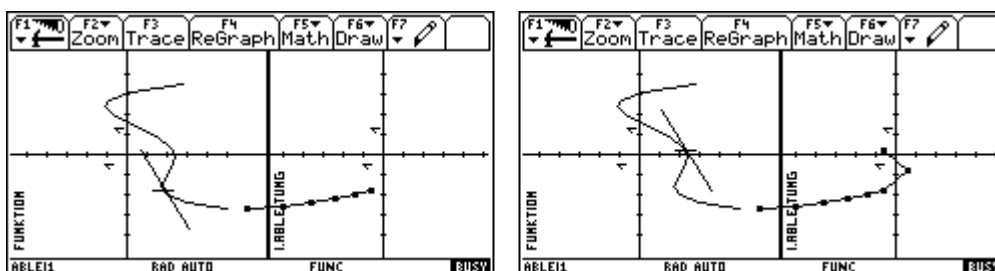
Das Menü erscheint. Wir wählen bei [F1] 1. Ableitung interaktiv.



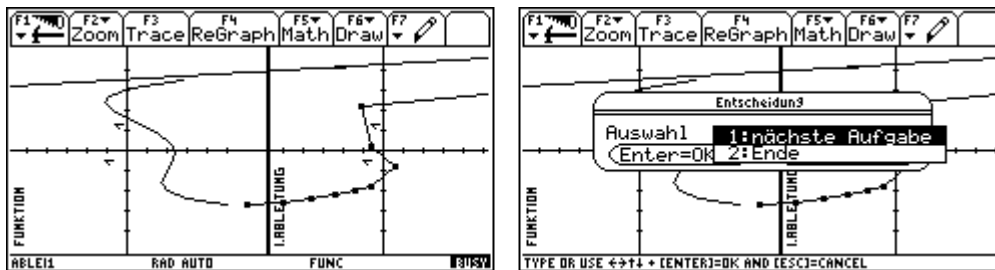
Der Menüpunkt Funktion aus der Bibliothek bedeutet, dass per Zufallszahl eine Polynomfunktion dritten oder vierten Grades aus einer Bibliothek von 50 solchen Funktionen ausgewählt wird. Die Funktionen der Bibliothek sind an die feste Größe des Bildschirmausschnittes der für die Darstellung der Graphen verwendet wird angepaßt. Der Bildschirm ist im Hochformat zu verwenden. Im oberen Teil wird der Graph der Funktion dargestellt. Am linken Ende des Graphen ist die Tangente dargestellt.



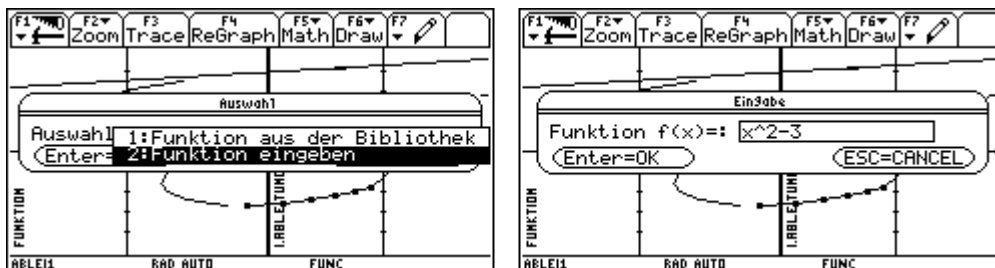
Mit \odot wandert die Tangente der Kurve entlang und ihre Anstiegswerte werden im unteren Teil des Bildschirmes dargestellt, damit Schritt für Schritt die 1. Ableitung entsteht.



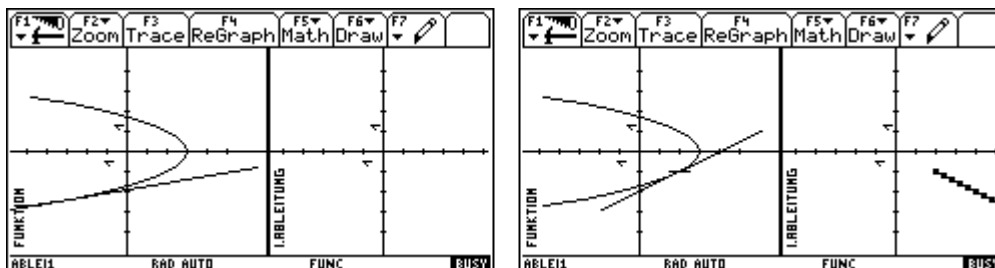
Erreicht man den rechten Rand des Bildschirmes, so wird der Programmteil beendet. Man hat die Wahlmöglichkeit zwischen Weitermachen und Aufhören. Mit $\boxed{2nd}\boxed{ESC}$ kann auch früher beendet werden.



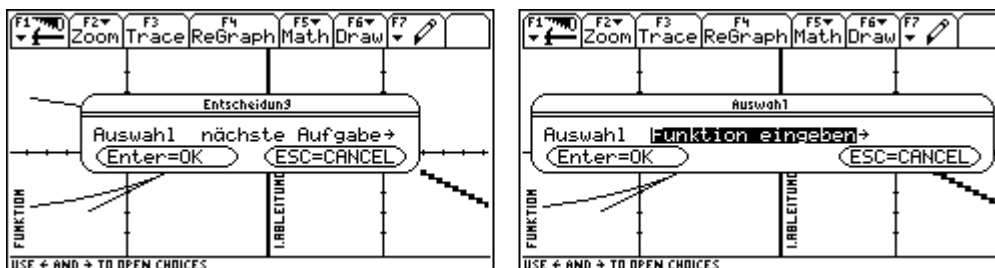
Diesmal geben wir die Funktion selbst ein. Das Programm ist für Polynomfunktionen mit Grad größer als 1 vorgesehen. Man kann es aber natürlich auch mit anderen Funktionen probieren. Dabei sollte man die Ausmaße des dargestellten Fensters berücksichtigen ($x_{min} = -4, x_{max} = 4, y_{min} = -5, y_{max} = 5$).



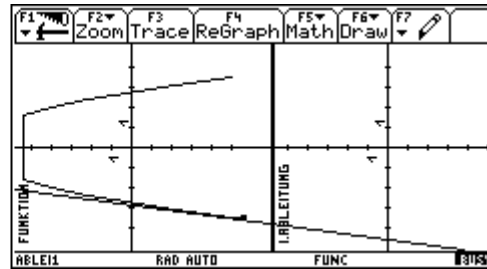
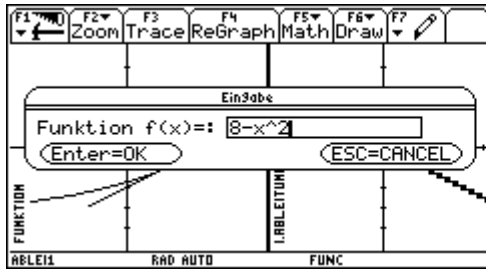
Die Schüler können also eigenständig Erkenntnisse über die 1. Ableitung sammeln.



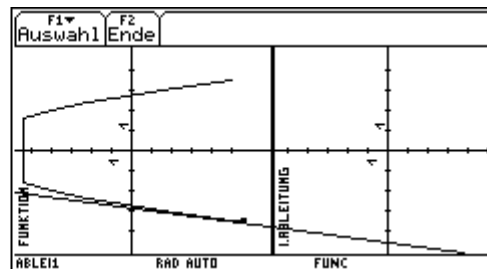
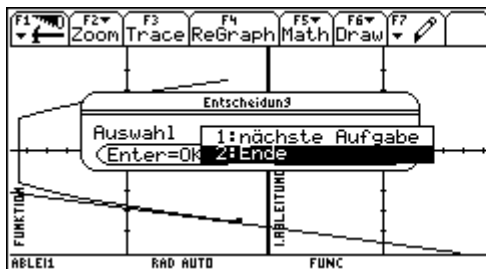
Zum Abschluß wollen wir uns noch ansehen, was passiert, wenn der Graph nicht in das Graphikfenster paßt.



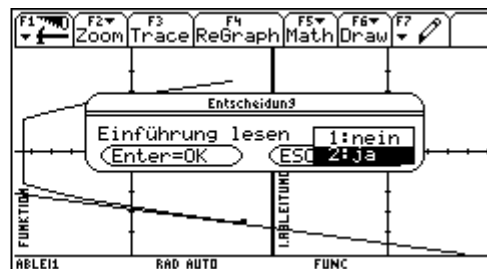
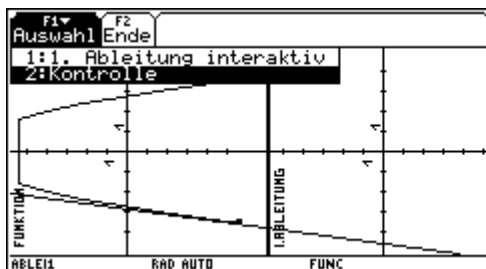
An den Rändern wird richtig abgeschnitten, aber sonstige außerhalb des Fensters liegende Graphenteile verursachen eine falsche Darstellung des Graphen.



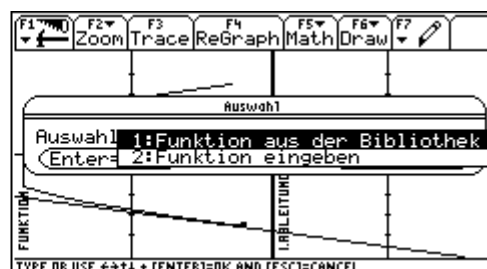
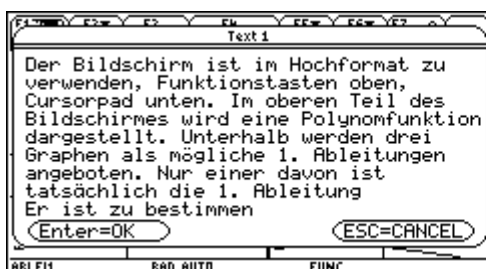
Am Ende gibt es die Auswahl zwischen dem Beenden und dem Fortsetzen mit einem neuen Beispiel. Mit **[F2]** wird die Programmgruppe verlassen.



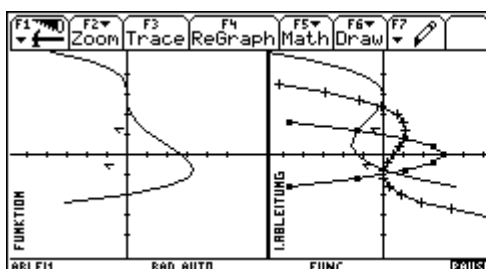
Nun kann mit dem Programm **[F1]** Kontrolle das Verständnis über die Zusammenhänge zwischen der Funktion und der 1. Ableitung überprüft werden. Eine Einleitung schildert die Programmbenützung.



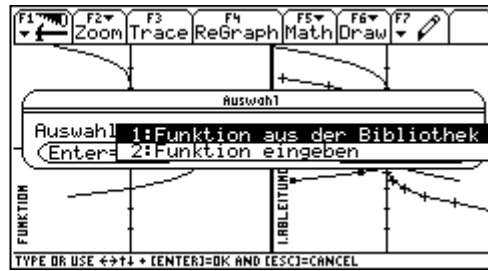
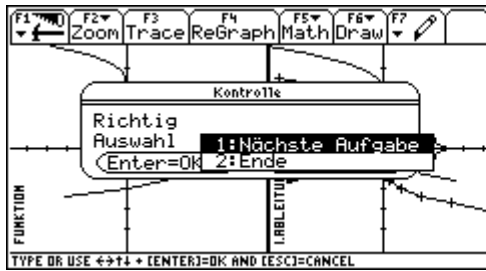
Wieder kann man zwischen Bibliothek und eigener Eingabe entscheiden.



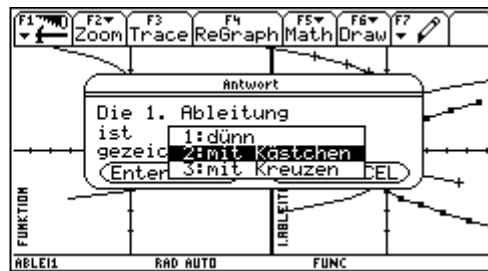
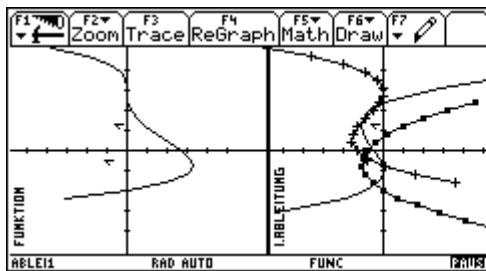
Hier wurde eine Funktion aus der Bibliothek gewählt.



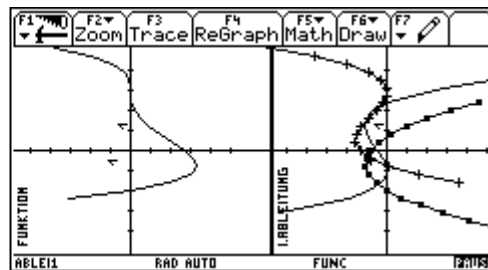
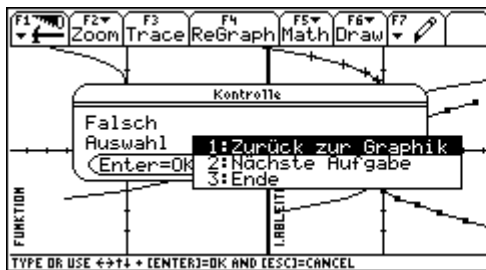
Das Programm kontrolliert die Richtigkeit der Antwort. Wir wählen ein weiteres Beispiel aus der Bibliothek.



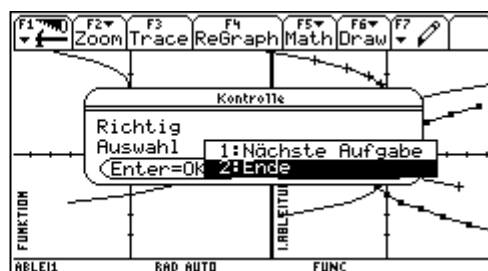
Diesmal wählen wir eine falsche Antwort.



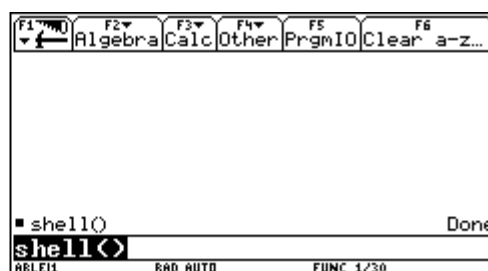
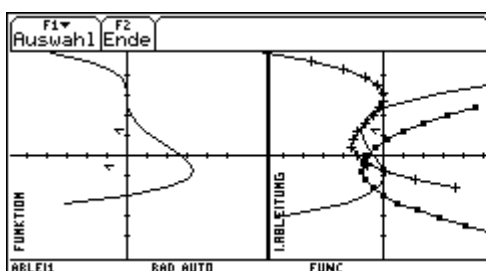
Nun gibt es die Möglichkeit zur Graphik zurückzukehren und den Fehler auszubessern.



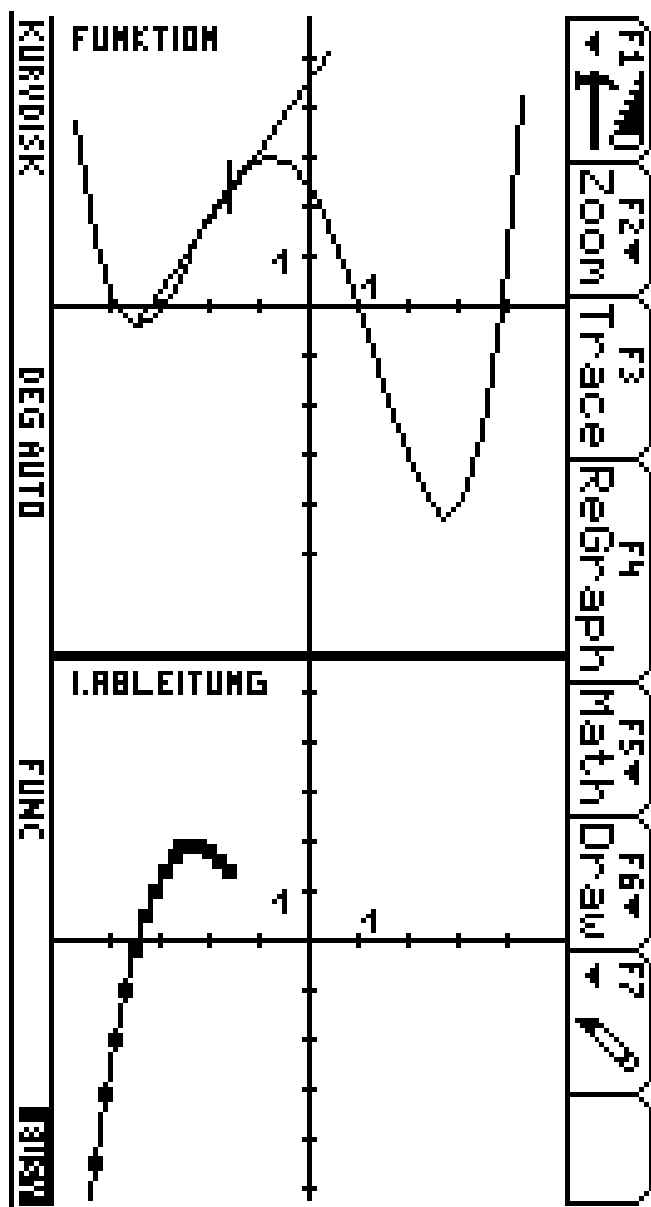
Dann wählen wir Ende.



Mit [F2] wird die Programmgruppe verlassen.



Zum Abschluß ein gedrehtes Bild des Programmbildschirmes.

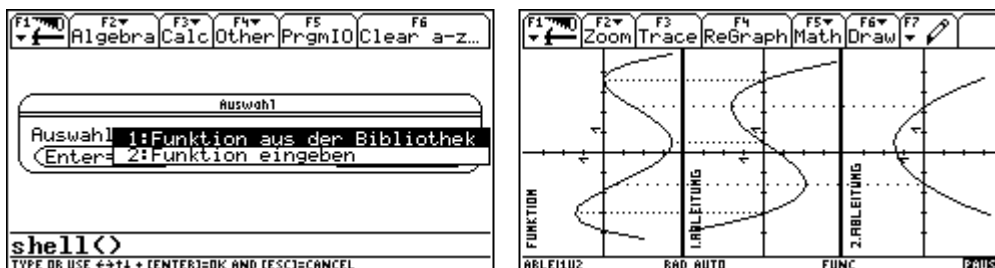


3. Programmgruppe: Ableit1u2

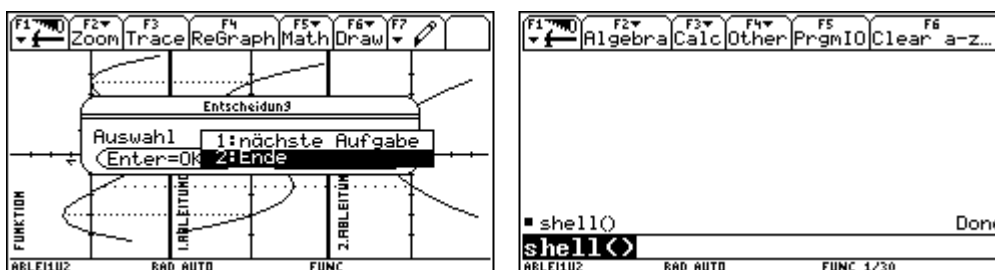
In den Folder ablei1u2 wechseln und das Programm shell() starten.



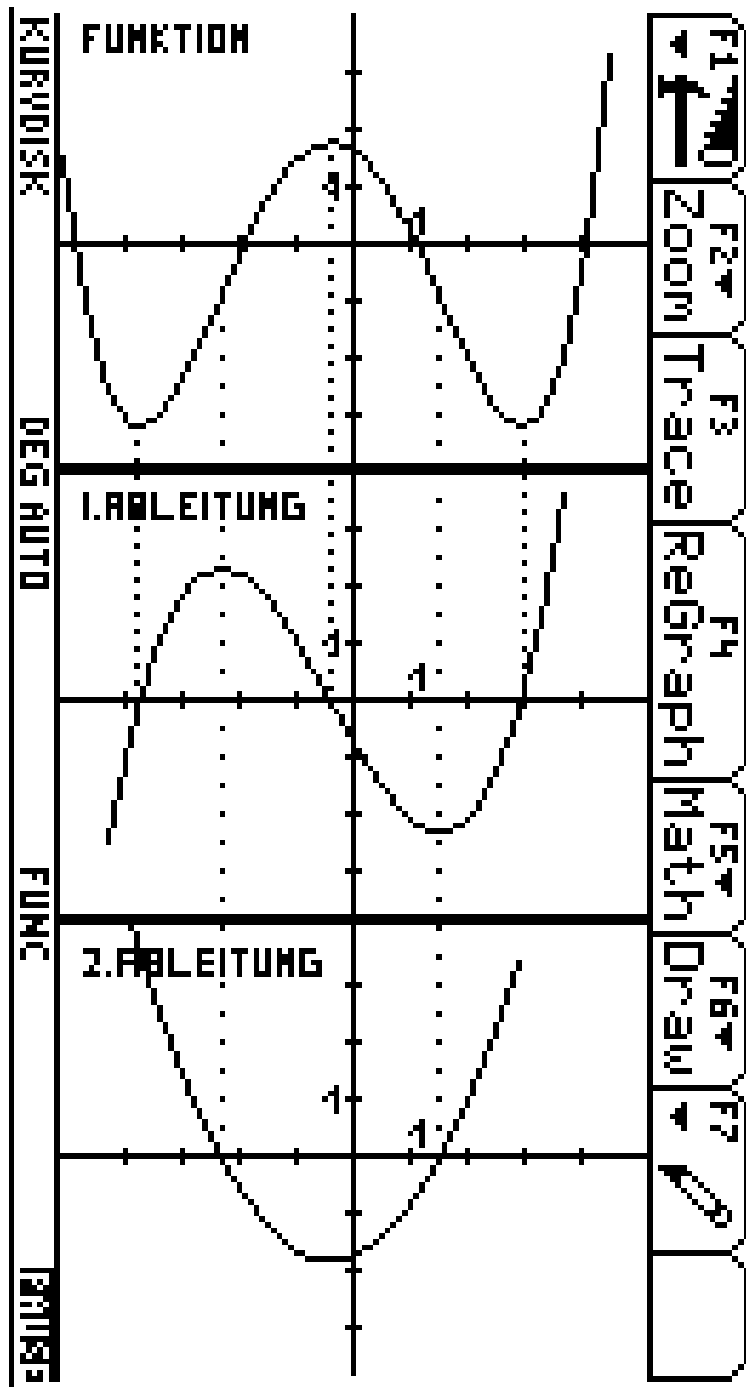
Wie in der Programmgruppe ablei1 kann eine Funktion aus der Bibliothek gewählt werden oder direkt eingegebene werden. Nimmt man keine Funktion aus der Bibliothek, so muß man wieder berücksichtigen, dass der Graph innerhalb des Fensters verläuft ($x_{min} = -4$, $x_{max} = 4$, $y_{min} = -4$, $y_{max} = 4$). Vorgesehen ist das Programm für Polynomfunktionen mit Grad größer als 1. Aber man kann es durchaus auch mit anderen Funktionen probieren. Der Bildschirm wird in 3 Teile geteilt und ist in Hochformat zu verwenden. Funktion, 1. Ableitung und 2. Ableitung sind übereinander gezeichnet. Die relativen Extremwerte der Funktion und die Nullstellen der 1. Ableitung, bzw. die Wendepunkte der Funktion und die Nullstellen der 2. Ableitung sind durch punktierte Linien miteinander verbunden.



Mit $\text{2nd}[\text{ESC}]$ wird das Beispiel beendet. Dann hat man die Wahlmöglichkeit zwischen dem Beenden und einem neuen Beispiel.



Abschließend noch eine gedrehte Abbildung des Programmbildschirmes.

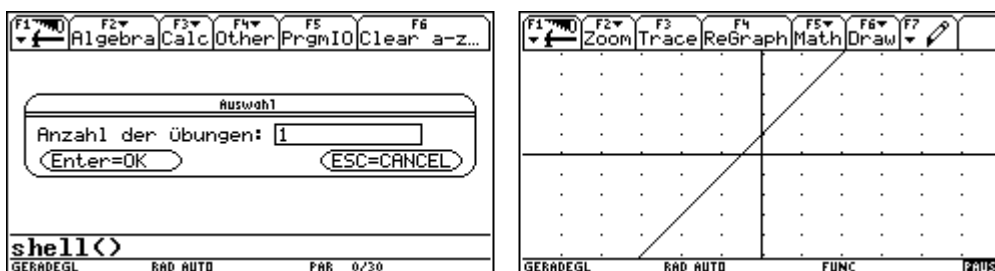


4. Programmgruppe: geradgl

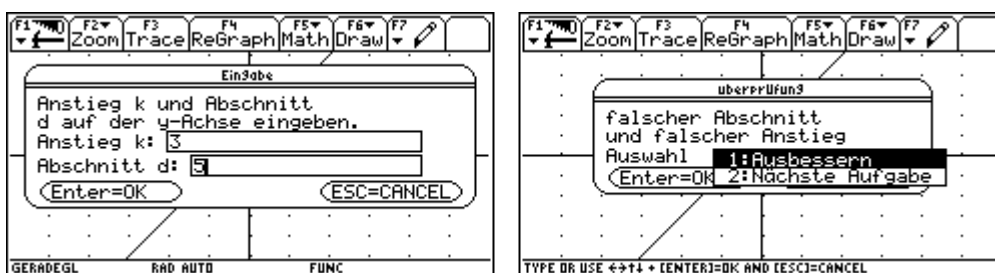
In den Folder geradgl wechseln und das Programm shell() starten.



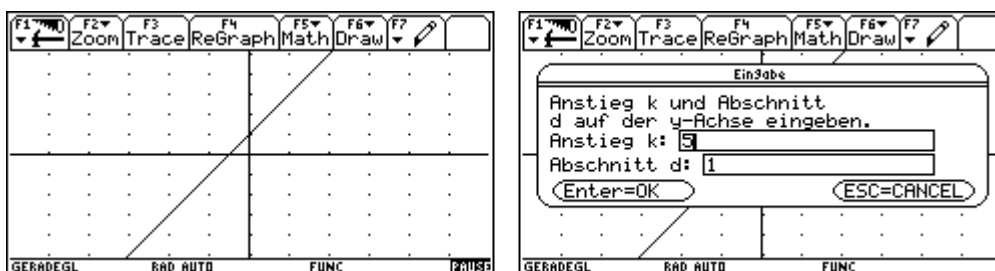
Die Anzahl der Übungen kann bestimmt werden. Der Graphikbildschirm mit dem Graphen einer linearen Funktion (durch Zufallszahlen bestimmt) erscheint ($x_{min} = -6, x_{max} = 6, y_{min} = -5, y_{max} = 5$). Die Abschnitte auf der y-Achse sind immer ganzzahlig, der Anstieg hat ganzzahlige Werte oder Bruchteile nicht kleiner als Viertel.



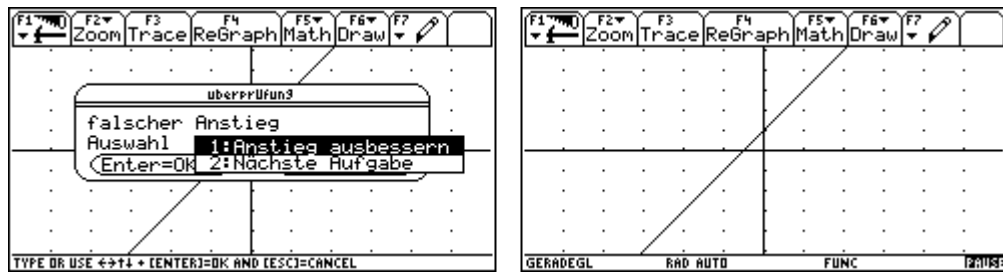
Nun wird man aufgefordert k und d einzugeben. Es erfolgt die Kontrolle mit der Möglichkeit auszubessern oder zum nächsten Beispiel gelangen bzw. nach dem letzten Beispiel zu beenden. Wir wählen ausbessern.



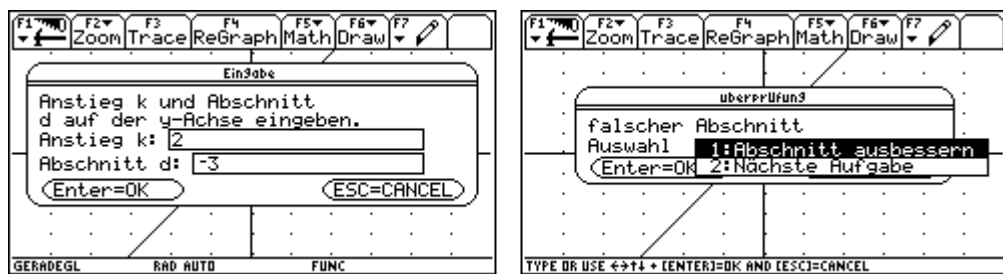
Damit kehrt man zur Graphik zurück und kann in der Folge die Werte ändern.



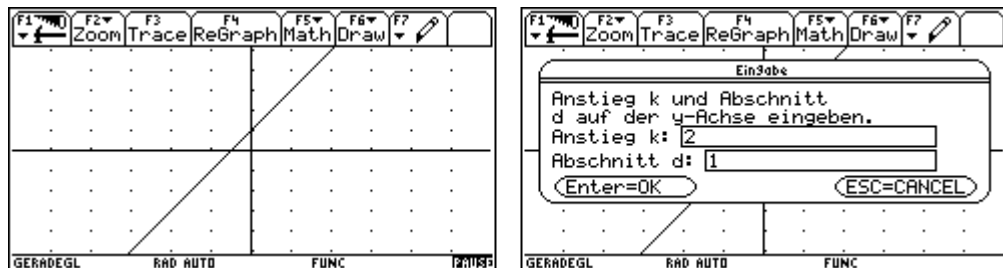
Ist k oder d richtig, so wird nur der andere Wert als falsch angezeigt. Wieder hat man die Möglichkeit auszubessern oder aufzuhören. Wir wählen ausbessern und kehren damit zur Graphik zurück.



In gleicher Weise verhält sich das Programm, wenn man den Anstieg richtig angibt und den Abschnitt falsch bestimmt. Siehe die kommenden Abbildungen.



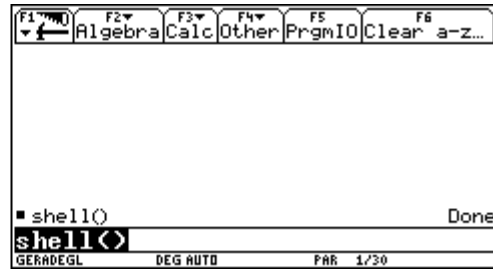
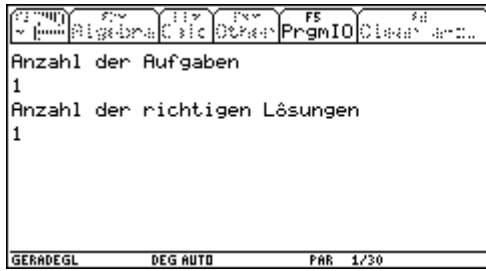
Nun geben wir beide Werte richtig ein.



Es erfolgt die Bestätigung und danach das nächste Beispiel bzw. nach dem letzten Beispiel eine Übersicht über die Aufgaben. Mit **ENTER** wird der Programm verlassen.



Mit **F5** kehrt man aus dem Programmbildschirm in den Homebereich zurück.

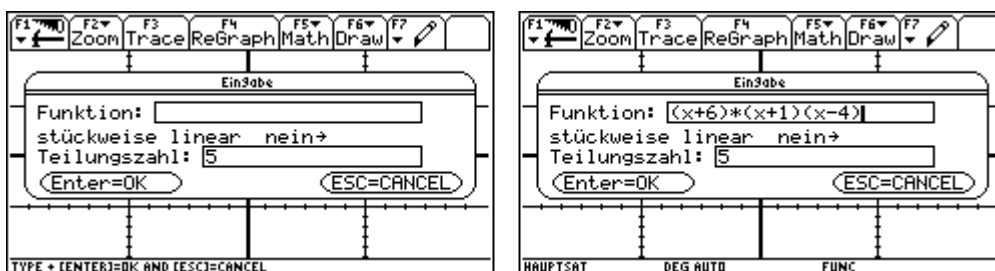


5. Programmgruppe:hauptsat

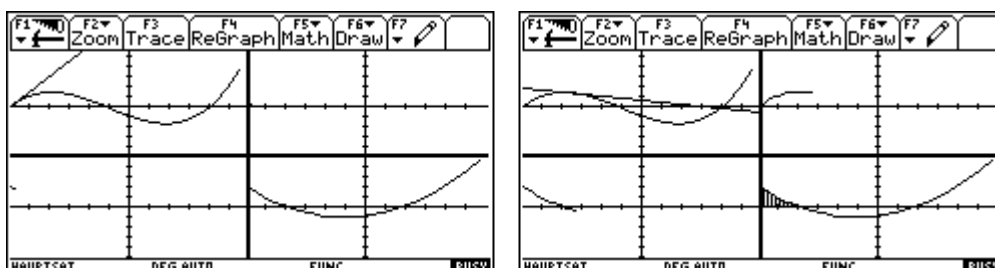
In den Folder hauptsat wechseln und das Programm shell() starten.

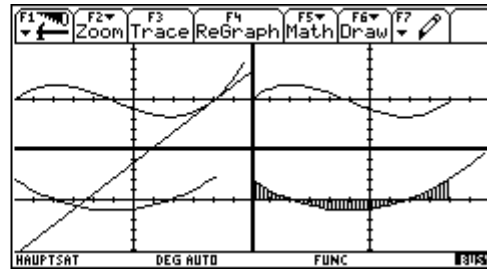
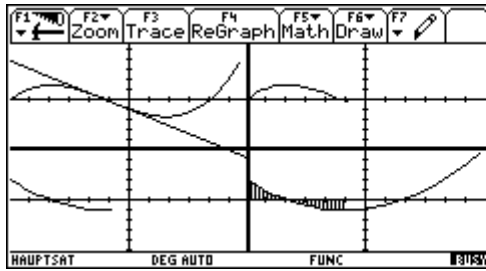


Die Funktion, an der der Hauptsatz der Differential- und Integralrechnung gezeigt werden soll, ist einzugeben. Programmiert wurde für Polynomfunktionen mit Grad größer 0. Zunächst stellen wir stückweise linear auf nein, womit auch die Teilungszahl bedeutungslos wird.

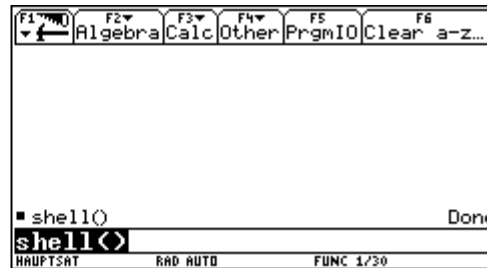
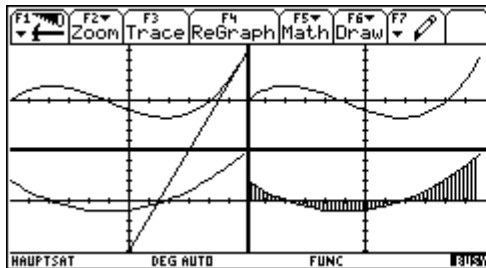


Der Bildschirm wird in vier Teile geteilt. Jeder Teil hat folgende Bildschirmausmaße: $x_{min} = -6$, $x_{max} = 6$, $y_{min} = -5$, $y_{max} = 5$) Links oben ist der Graph der Funktion abgebildet, wobei die Funktion im y-Bereich so gestaucht wird, dass der Graph in das Fenster ($y_{min} = -5$ und $y_{max} = 5$) hineinpasst. Außerdem wird der Graph so verschoben, dass er immer bei $x = -6$ eine Nullstelle besitzt. Am linken Rand des Graphen ist die Tangente eingezeichnet. Rechts unten ist der Graph der 1. Ableitung der gestauchten Funktion abgebildet. Mit \odot wandert die Tangente am Graphen der Funktion nach rechts und die Anstiege der Tangente werden links unten dargestellt, womit Schritt für Schritt die 1. Ableitung entsteht. Gleichzeitig wird die Fläche unter der 1. Ableitung schrittweise von links nach rechts markiert und der Wert der jeweiligen Fläche rechts oben dargestellt, womit Schritt für Schritt rechts oben ein unbestimmtes Integral entsteht. Damit läßt sich der Hauptsatz für verschiedene Funktionen erläutern.

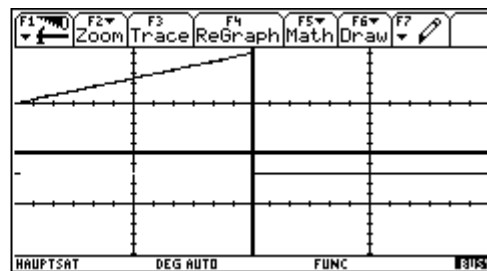
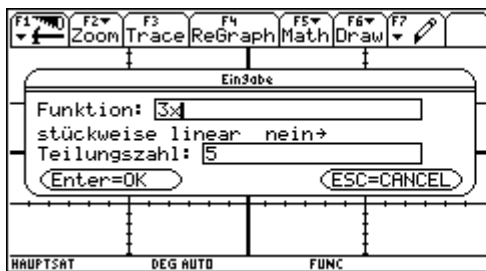




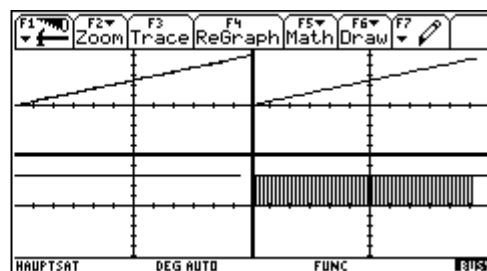
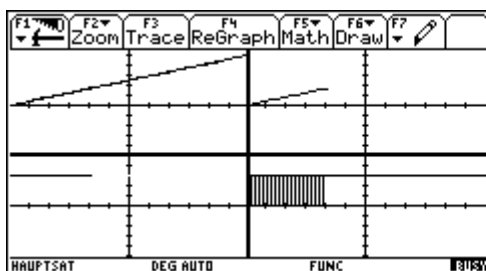
Mit $\text{2nd}[\text{ESC}]$ wird das Programm beendet.



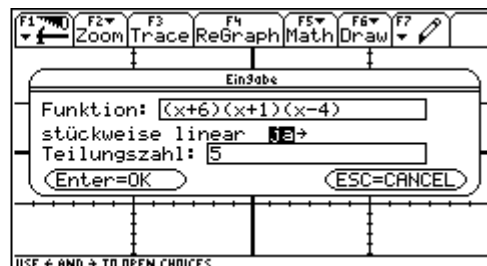
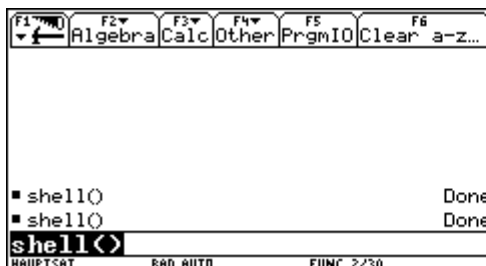
Das Programm dient nun aber auch dazu den Hauptsatz zu verstehen. Dazu verwenden wir zunächst einmal eine lineare Funktion. Stückweise linear belassen wir auf nein.



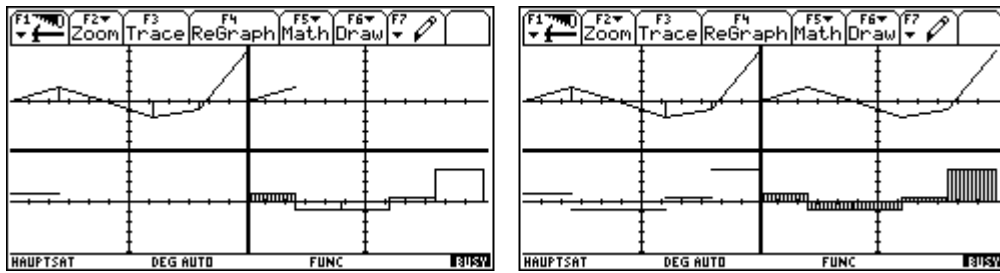
Hier ist es einsichtig oder sehr leicht zu zeigen, dass die Flächenberechnung die Umkehrung der Berechnung des Anstieges der Tangenten ist.



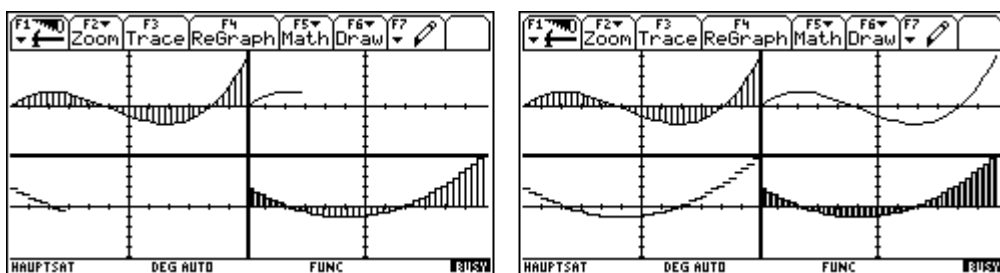
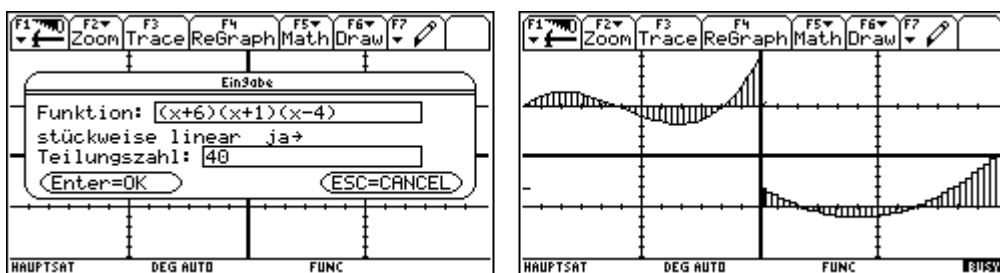
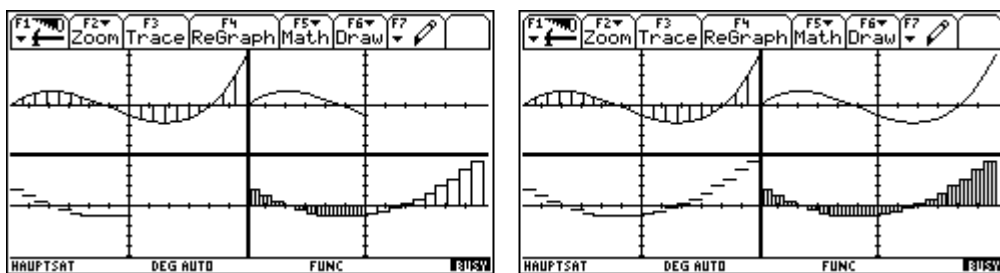
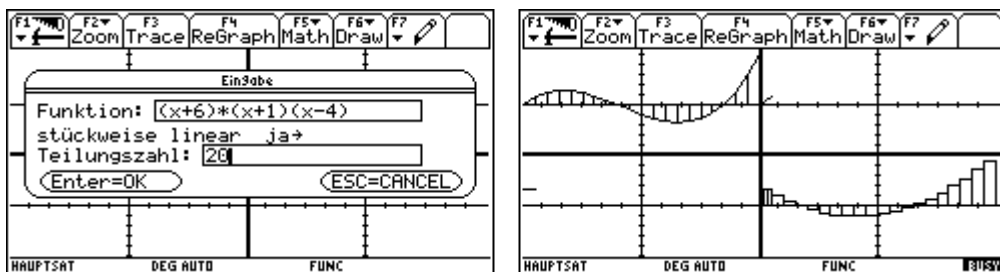
Nun kehren wir zu unserer ursprünglichen Funktion zurück und stellen stückweise linear auf ja und die Teilungszahl auf 5.



Das sichtbare Intervall wird in 5 gleiche Teile geteilt und die Funktion durch die entsprechenden Sekantenstücke ersetzt. Auch für diesen Fall läßt sich nun der Hauptsatz gut erklären.



Durch Erhöhen der Teilungszahl kann man nun die Funktion und ihre Ableitung optisch beliebig genau annähern, wobei die Erkenntnisse der letzten Schritte erhalten bleiben. Es kann also ein tieferes Verständnis auch ohne exaktem Beweis erreicht werden.



6. Programmgruppe: Integral

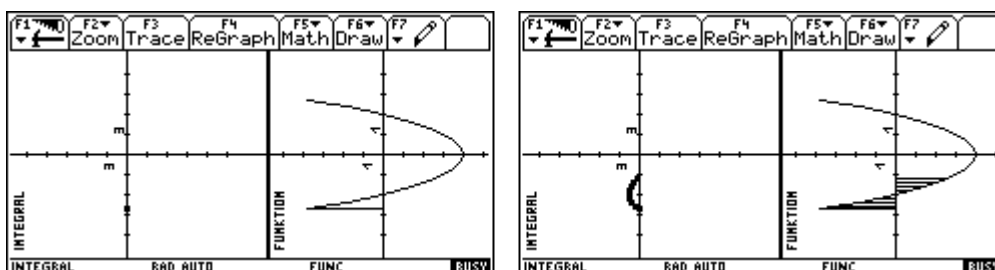
In den Folder integral wechseln und das Programm shell() starten.



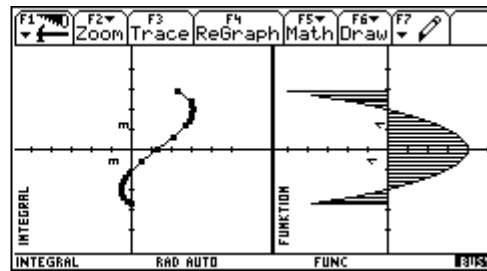
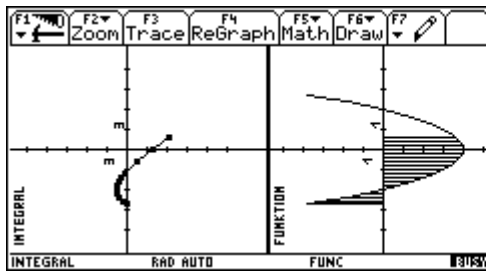
Wie schon in der Programmgruppe ablei1 beschrieben kann zwischen Bibliothek und eigener Funktion ausgewählt werden. Wir geben eigenständig eine Parabel ein.



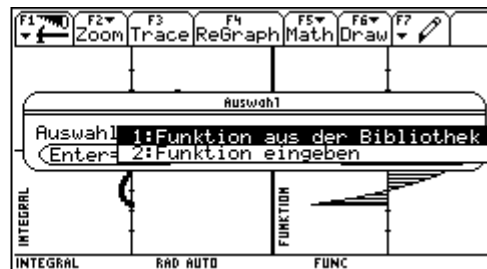
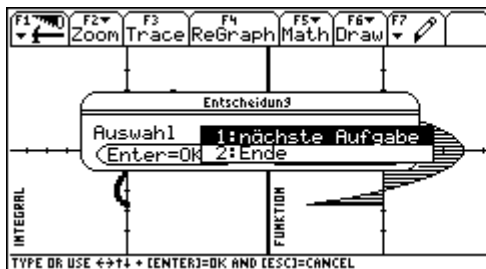
Der Bildschirm ist im Hochformat zu verwenden. Im unteren Teil ist der Graph der Funktion abgebildet. Da auch hier die Ausmaße des Bildschirms ($x_{min} = -5$, $x_{max} = 5$, $y_{min} = -6$, $y_{max} = 6$) nicht verändert werden können, gelten für eigenständig eingegebene Funktionen die gleichen Einschränkungen, wie für die Programmgruppe ablei1 beschrieben. Mit \odot wird die Fläche unter der Kurve schrittweise von links nach rechts markiert und der Wert der Fläche im oberen Teil des Bildschimes dargestellt, womit ein unbestimmtes Integral der Funktion entsteht. Interessant ist die Interpretation der Nullstellen. Sie führen zu einem Vorzeichenwechsel der Fläche und damit zu einem relativen Extremwert im unbestimmten Integral.



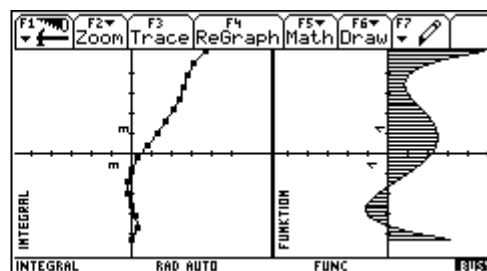
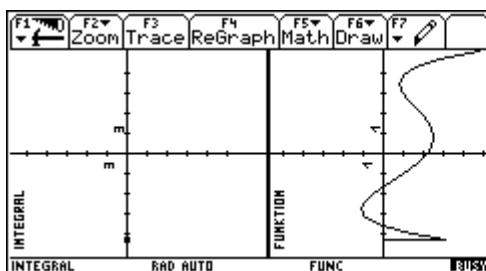
Mit $\boxed{2nd}$ \odot wird die Schrittweite des Verfahrens vergrößert.



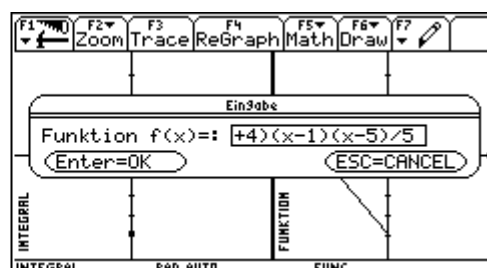
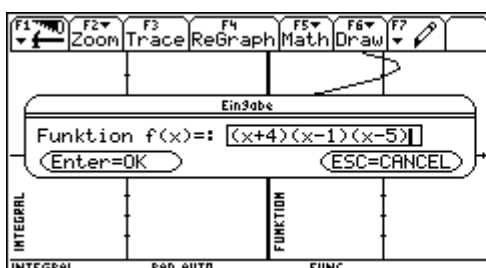
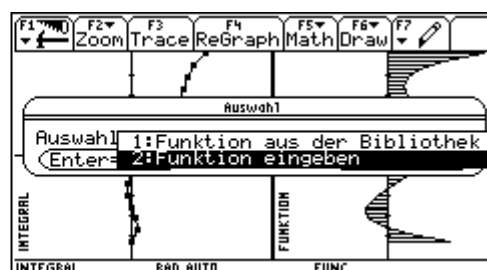
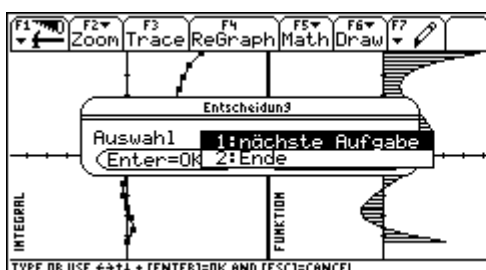
Mit $\boxed{2nd}\boxed{ESC}$ wird beendet. Dann gibt es die Wahlmöglichkeit für neue Aufgaben oder die Programmgruppe zu verlassen. Wir geben eine neue Funktion ein.



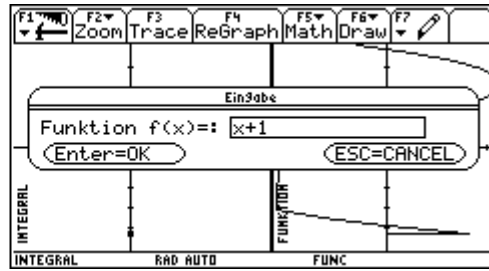
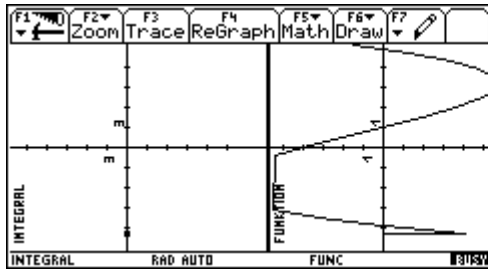
Diesmal wählen wir eine aus der Bibliothek.



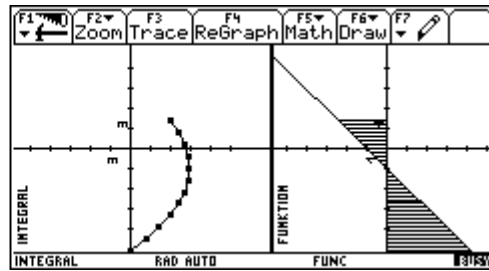
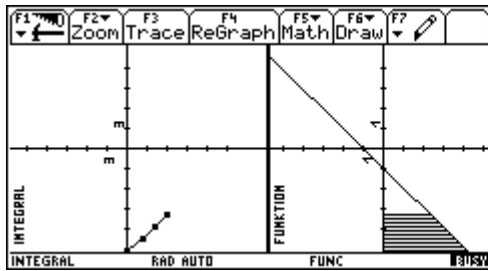
Nun geben wir eigenständig eine Funktion ein, deren Graph nicht in das Graphifenster paßt.



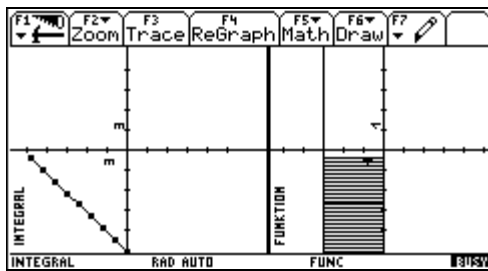
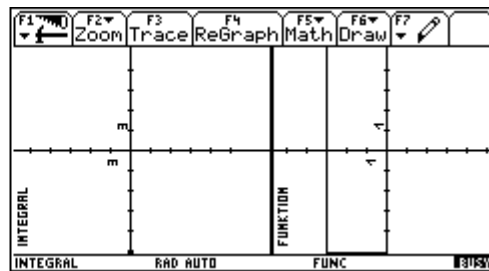
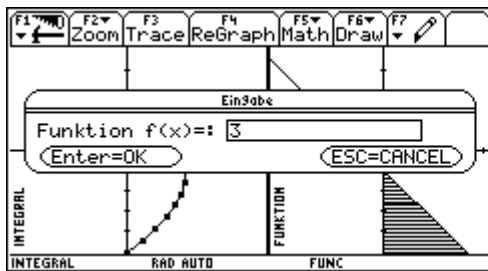
Der Graph wird fehlerhaft dargestellt. Nun geben wir eine lineare Funktion ein.



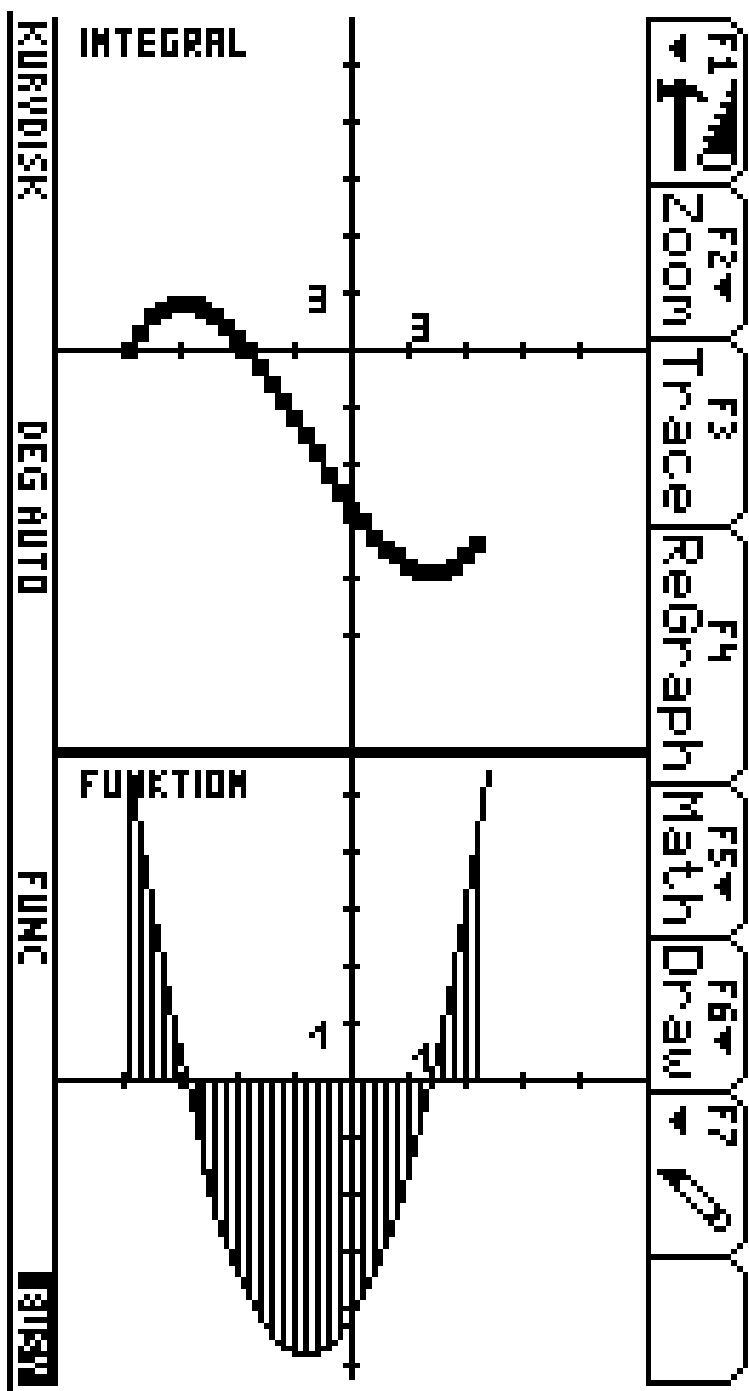
Eine Möglichkeit für Schüler sich an den Hauptsatz heranzutasten.



Auch für eine konstante Funktion ist das Ergebnis lehrreich.

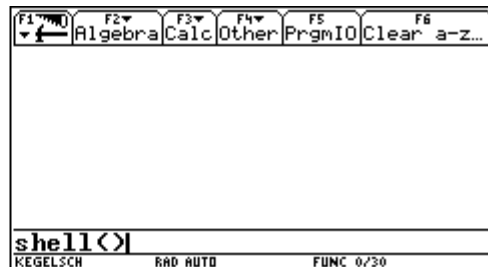


Zum Abschluß eine gedrehte große Abbildung des Programmbildschirmes.

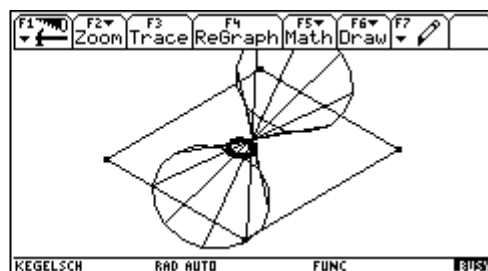
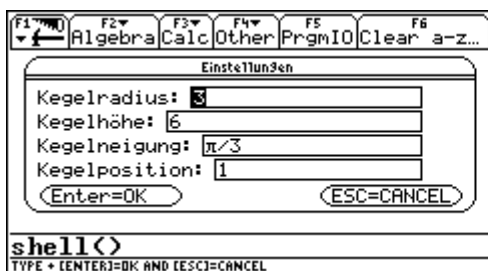


7. Programmgruppe: Kegelsch

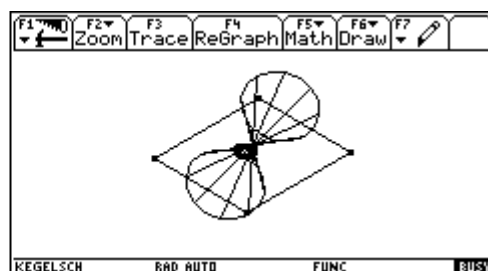
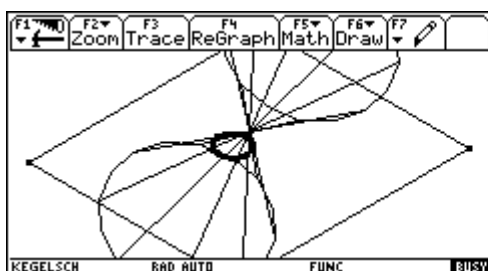
In den Folder kegelsch wechseln und das Programm shell() starten.



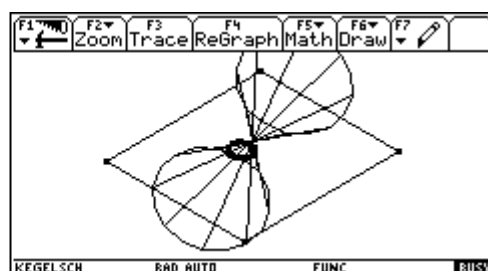
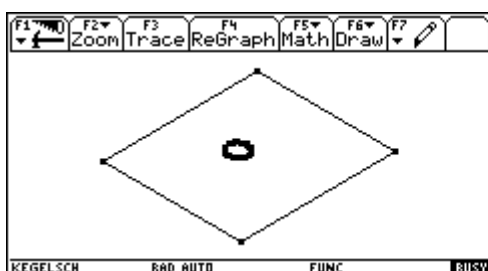
Die Achse des Kegels befindet sich in der yz-Ebene. Die Kegelneigung ist der Winkel zwischen Kegelachse und der y-Achse in Radiant. Die Kegelposition ist der Abschnitt, den die Kegelachse auf der z-Achse erzeugt. Geschnitten wird der Kegel mit der xy-Ebene. Dargestellt wird der Kegel, die Schnittebene und die Schnittfigur in Axiometrie. Der Kegelradius ist der Radius des Basiskreises, die Kegelhöhe bezieht sich auf einen Teil des Doppelkegels. Betrachten wir zunächst die Darstellung für die vorgegebenen Werte.



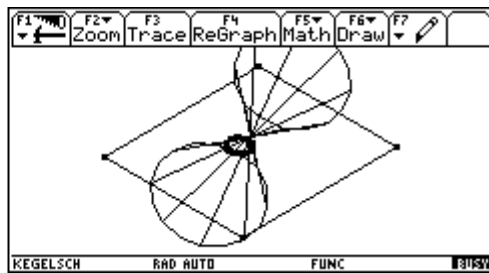
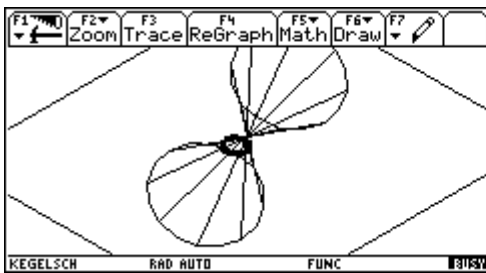
Mit der Taste k kann der Bildschirmausschnitt verkleinert, mit der Taste g vergrößert werden



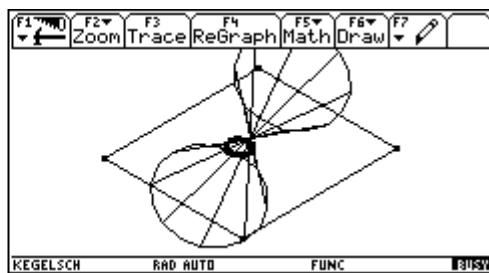
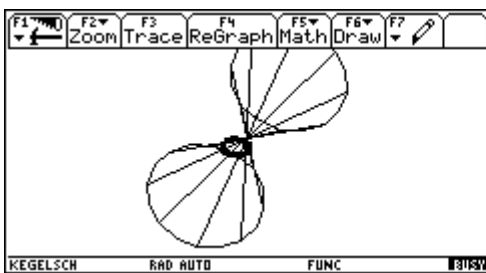
Mit der Taste l kann der Kegel ausgeblendet werden, mit \uparrow l kann er wieder eingeblendet werden.



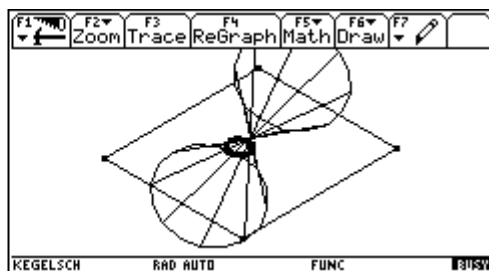
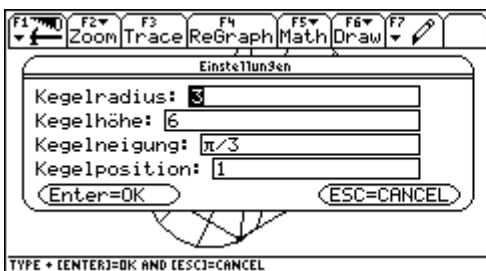
Mit der Taste w kann die Darstellung der xy-Ebene vergrößert werden, mit \uparrow w kann die Darstellung verkleinert werden.



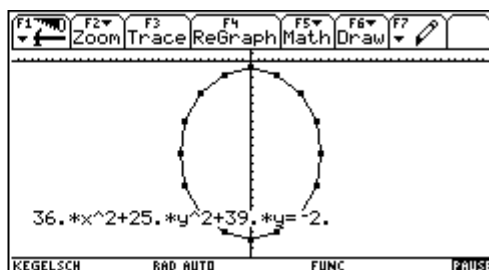
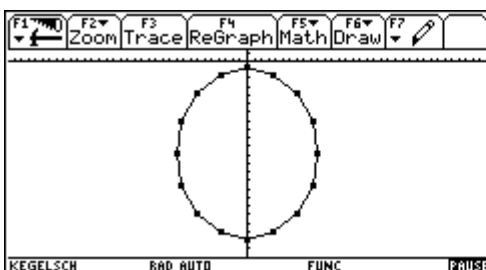
Damit kann man auch ein Ausblenden der Darstellung der xy-Ebene erreichen.



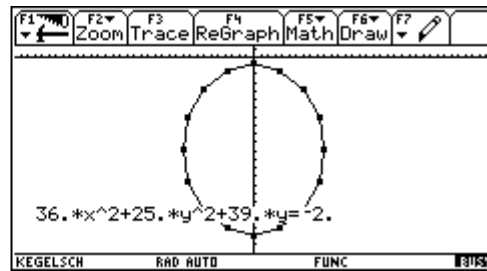
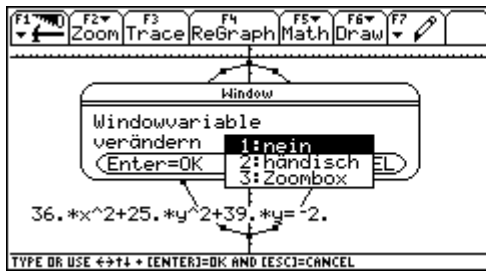
Mit der Taste e kann man eine Dialogbox aufrufen, in der alle Grundeinstellungen verändert werden können.



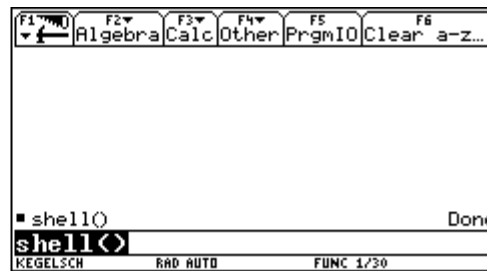
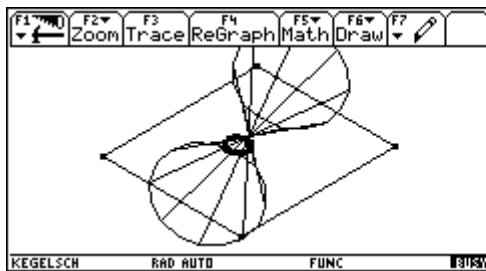
Mit der Taste x wird die xy-Ebene in Hauptlage mit dem Kegelschnitt gezeigt. Mit \uparrow wird die Gleichung eingeblendet. Die Koeffizienten sind auf ganze Stellen gerundet.



Mit \uparrow folgt eine Dialogbox, mit der die Windowvariablen für die Darstellung des Kegelschnittes geändert werden können.

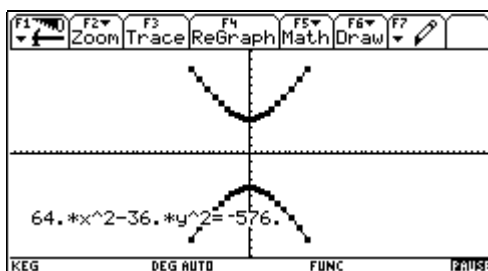
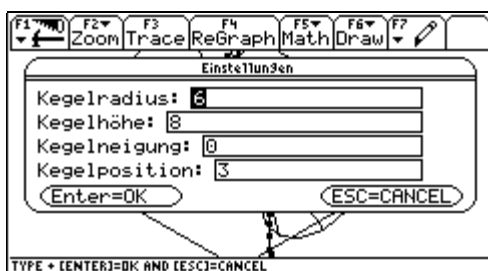


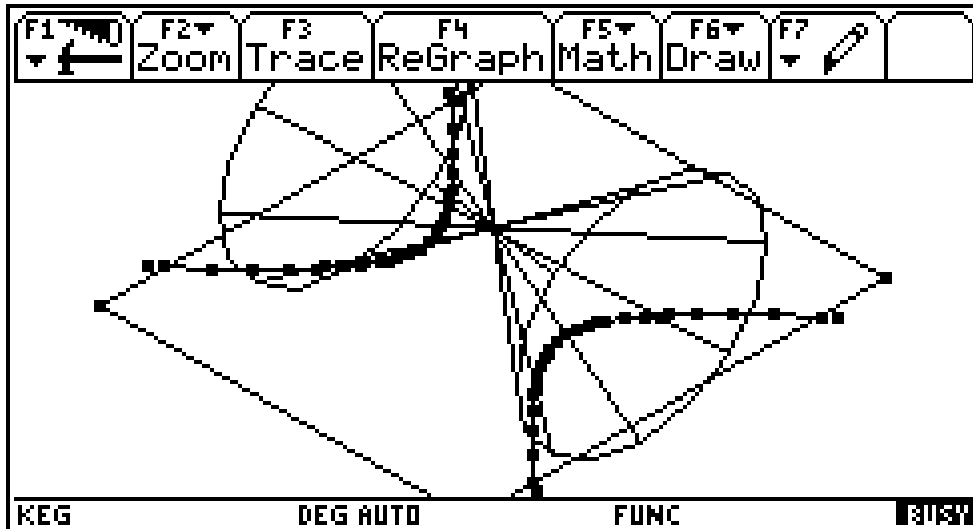
Mit der Taste y kehrt man zur 3-dimensionalen Darstellung zurück. Mit $2^{nd}[ESC]$ wird das Programm geändert.



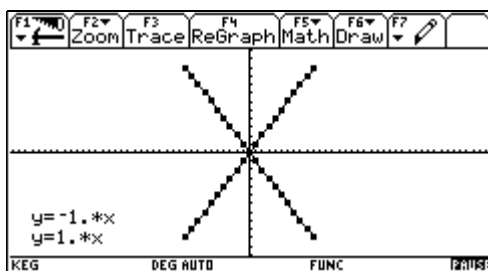
Es folgt ein Überblick über die möglichen Schnittvarianten.

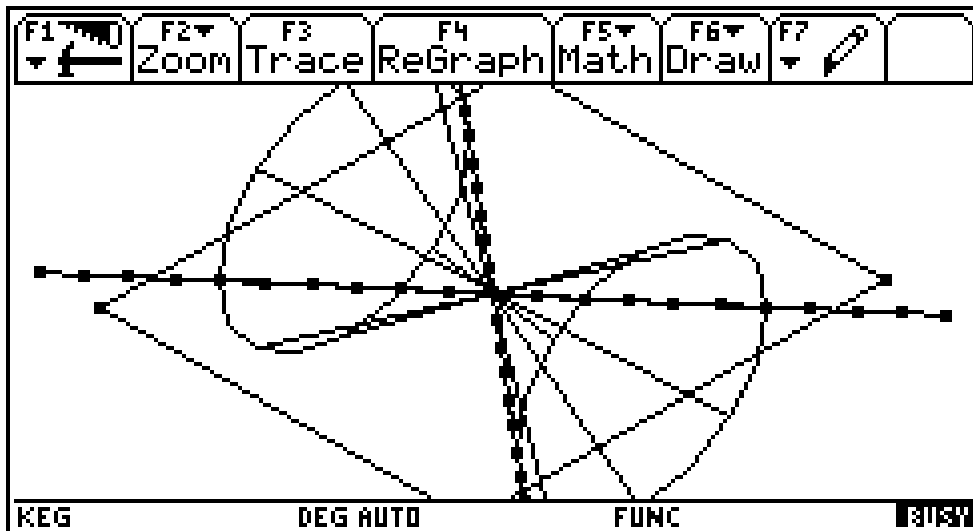
Hyperbel: Der Winkel zwischen der Schnittebene und der Kegellachse ist kleiner als der Winkel zwischen einer Erzeugenden und der Achse. Die Spitze liegt nicht in der Ebene.



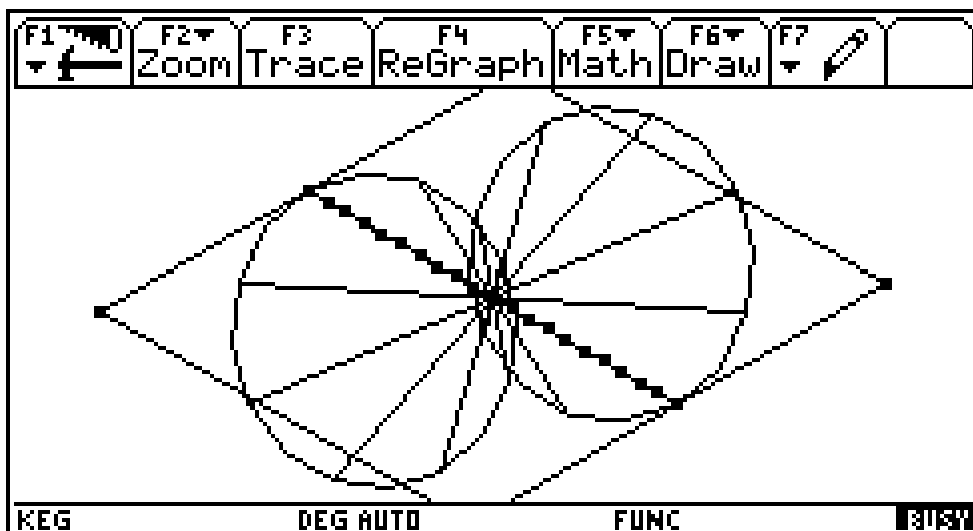
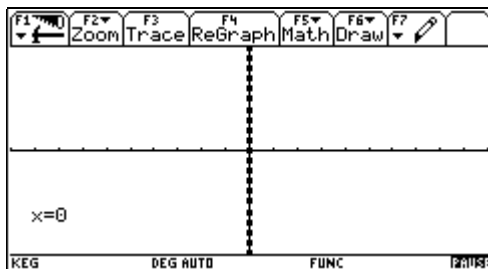
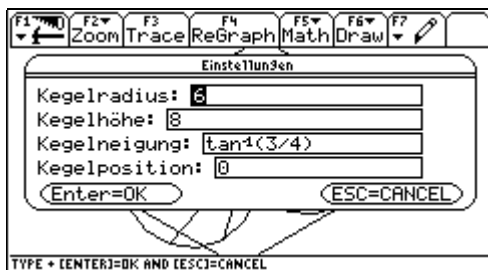


Zwei Gerade: Der Winkel zwischen der Schnittebene und der Kegelachse ist kleiner als der Winkel zwischen einer Erzeugenden und der Achse. Die Spitze des Kegels liegt in der Ebene. Zum Beispiel: die Achse liegt in der Ebene

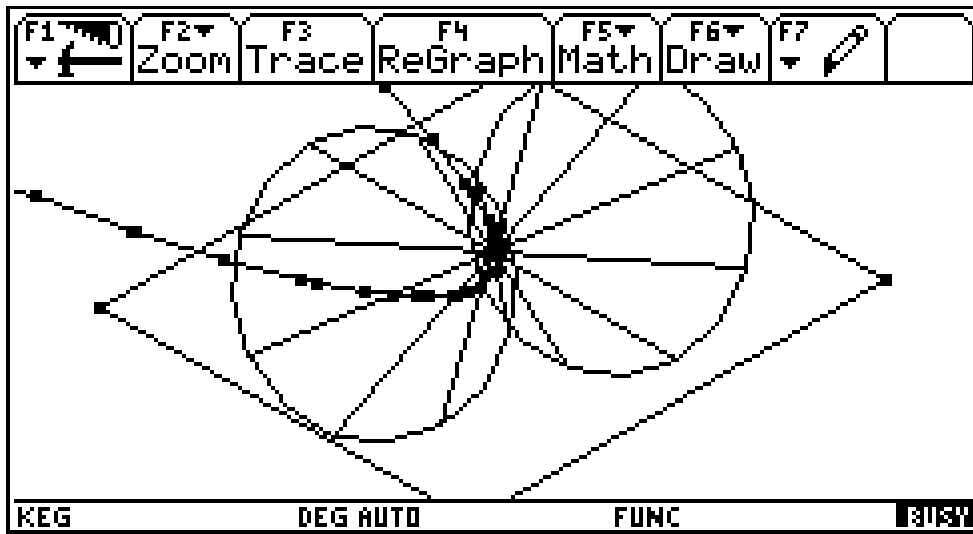




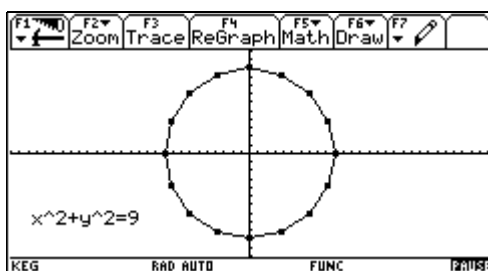
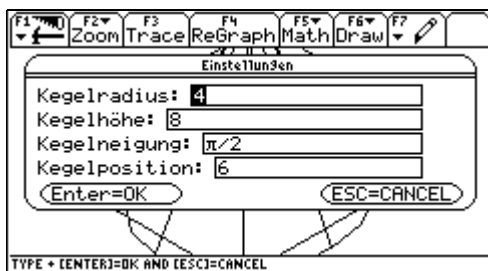
eine Gerade: die Ebene berührt den Kegel

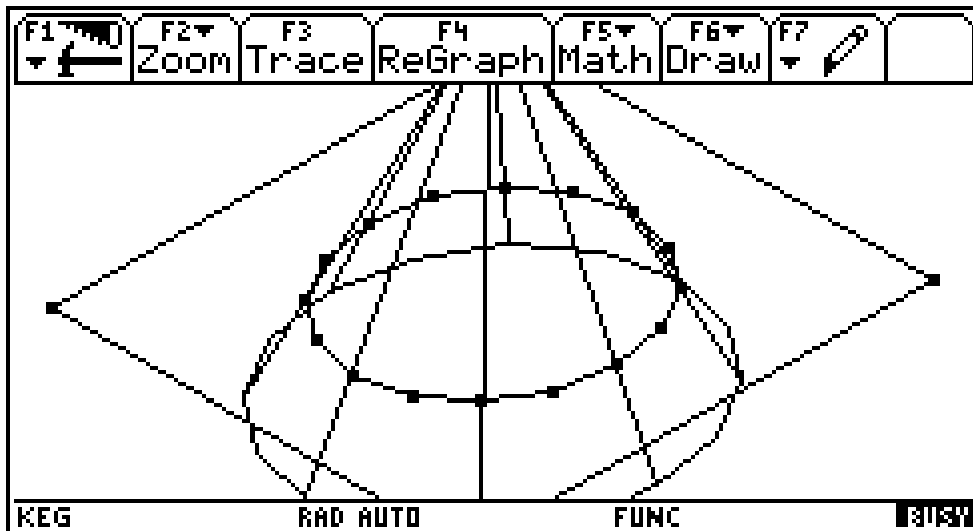


eine Parabel: die Schnittebene ist parallel zu einer Erzeugenden

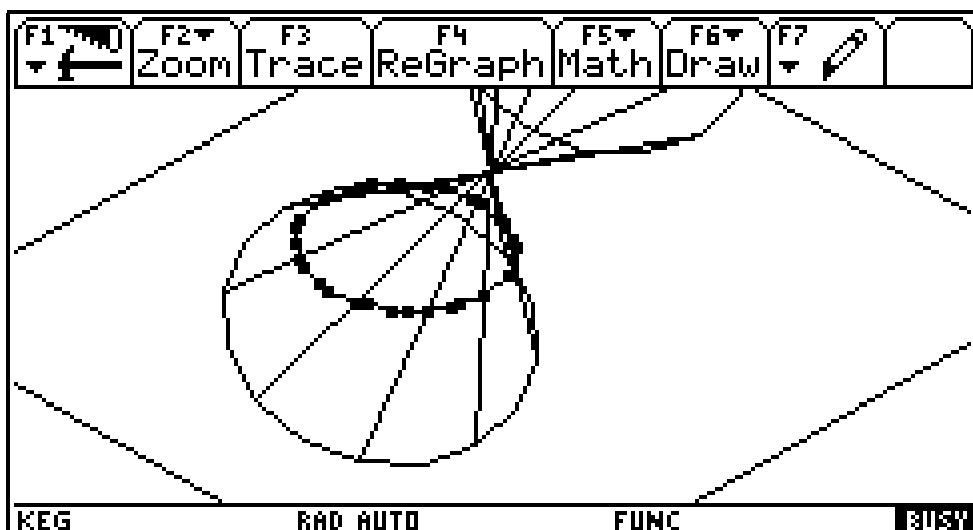
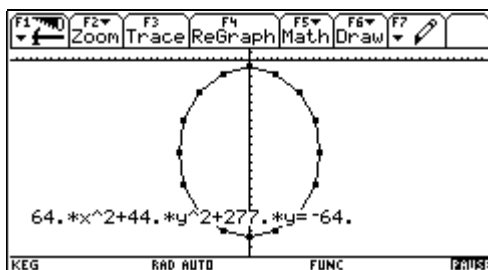
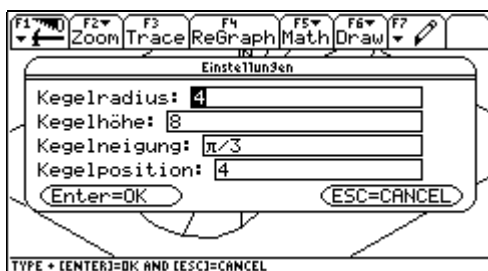


ein Kreis: Die Achse ist normal zur Schnittebene und die Spitze liegt nicht in der Ebene.

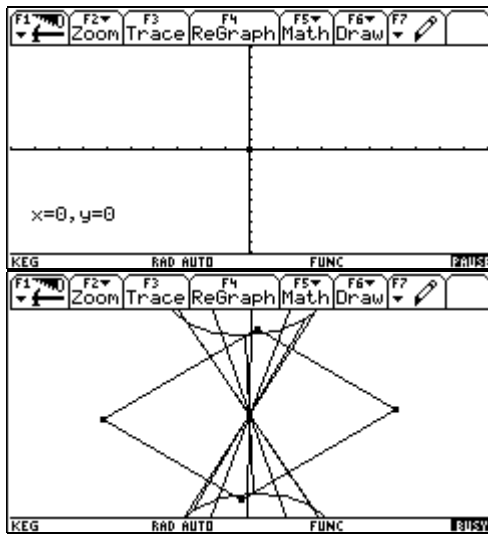
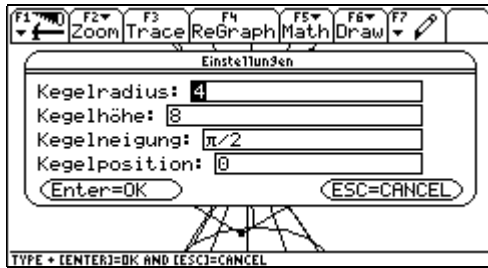




eine Ellipse: Der Winkel zwischen der Schnittebene und der Kegelachse ist größer als der Winkel zwischen einer Erzeugenden und der Kegelachse. Die Spitze liegt nicht in der Ebene.



ein Punkt: Die Spitze liegt in der Ebene und der Winkel zwischen der Schnittebene und der Kegelschneidung ist größer als der Winkel zwischen einer Erzeugenden und der Achse.

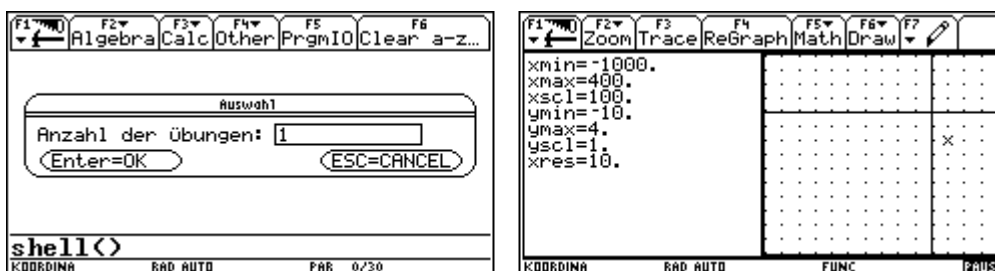


8. Programmgruppe: koordina

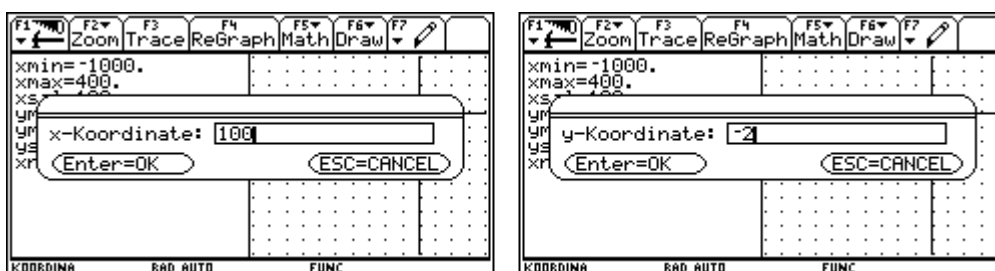
In den Folder koordina wechseln und das Programm shell() starten.



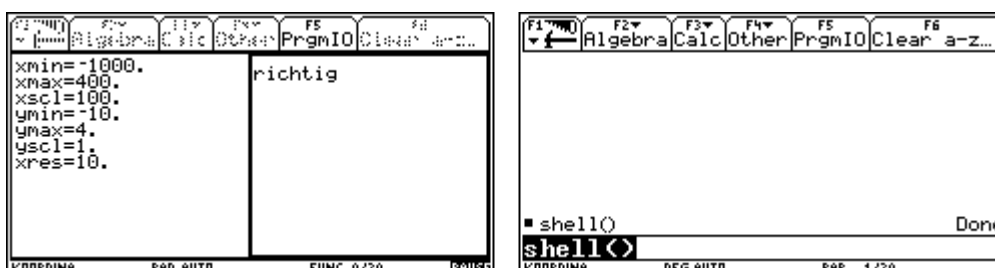
Anzahl der Übungen eingeben. Der Bildschirm wird geteilt. In der linken Hälfte sind die Windowvariablen zu sehen, in der rechten Hälfte das dazugehörige Koordinatensystem mit einem Gitter. Einer der Gitterpunkte ist markiert. Sowohl der Gitterpunkt, als auch die Ausmaße und Einheiten des Koordinatensystems werden durch Zufallszahlen bestimmt.



Das Programm fordert die Eingabe von x- und y-Koordinate des Punktes.



Und kontrolliert die Richtigkeit der Eingabe

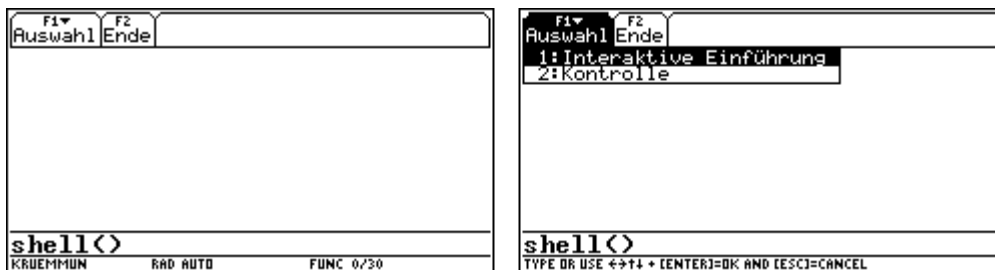


9. Programmgruppe: kruemmun

In den Folder kruemmun wechseln und das Programm shell() starten.



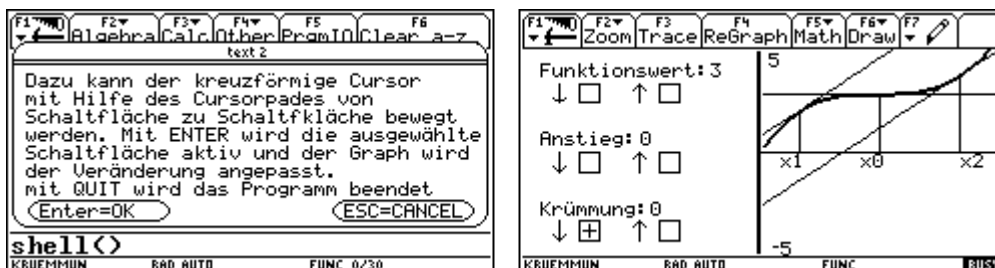
Bei [F1] Interaktive Einführung wählen.



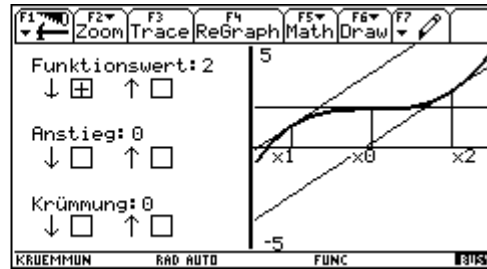
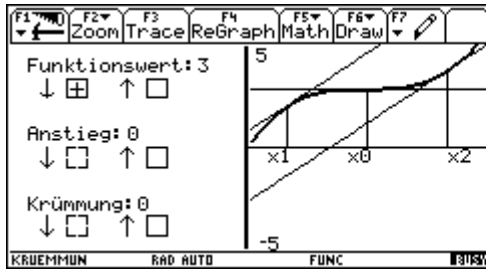
Es folgt die Möglichkeit eine Beschreibung dieses Programmteiles zu lesen. Die dargestellte Funktion ist eine Polynomfunktion folgender Bauart: $f(x)=x^3+bx^2+cx+d$, sodass die Funktion durch Funktionswert, 1. u. 2. Ableitung an der Stelle 0 bestimmt ist.



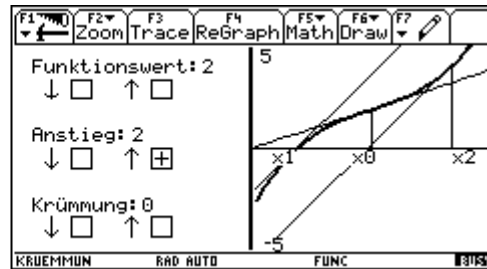
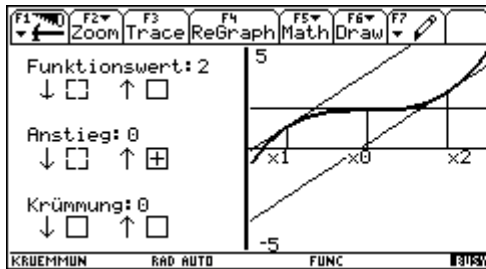
Danach startet das Programm.



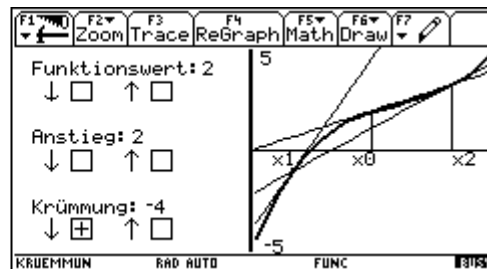
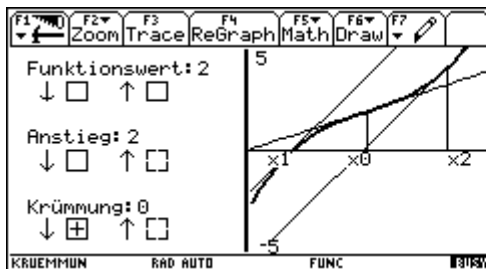
In den nächsten Abbildungen wurde der Funktionswert vermindert.



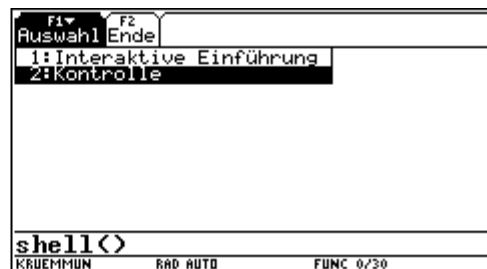
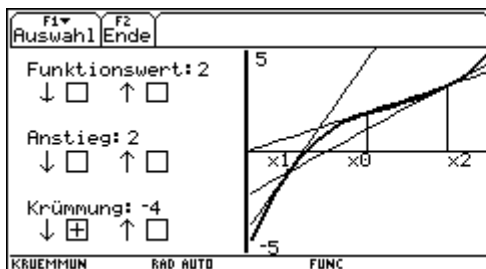
In den nächsten Abbildungen wurde der Anstieg vergrößert.



In den nächsten Abbildungen wird die Krümmung verkleinert.



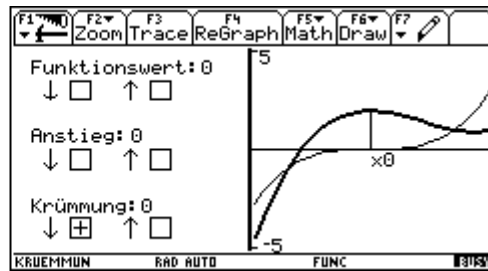
Mit $\boxed{2nd}\boxed{ESC}$ verlassen wir diesen Programmteil und wenden uns unter $\boxed{F1}$ der Kontrolle zu.



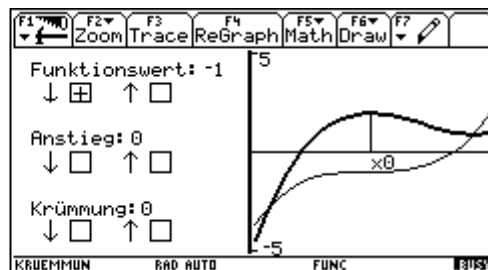
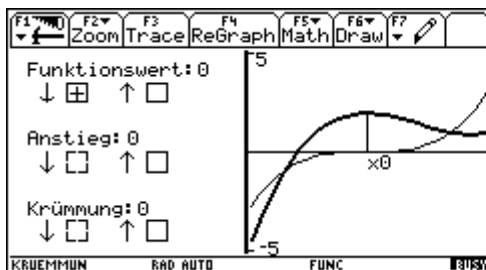
Wieder gibt es die Möglichkeit eine Programmbeschreibung zu lesen.



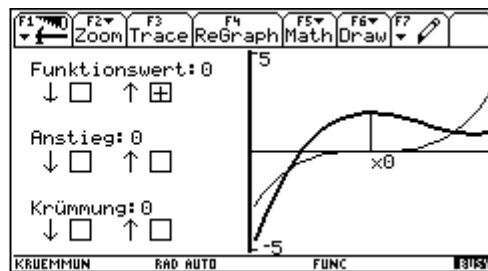
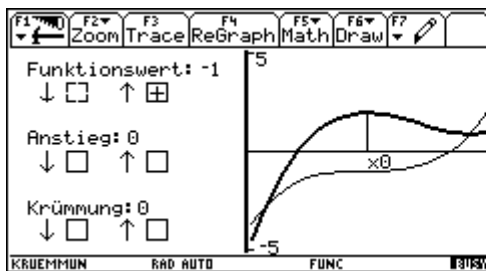
Am Ende der Einführung startet automatisch das Programm.



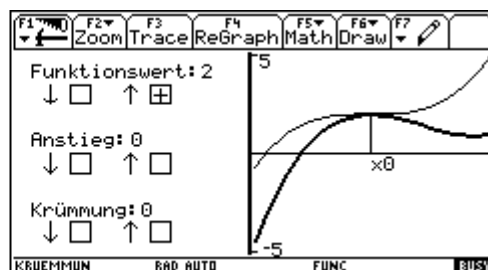
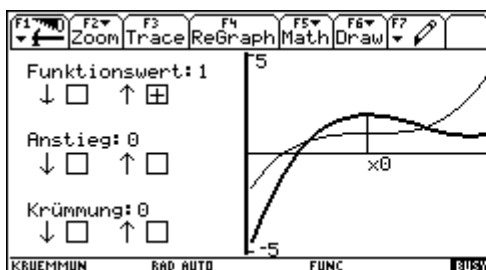
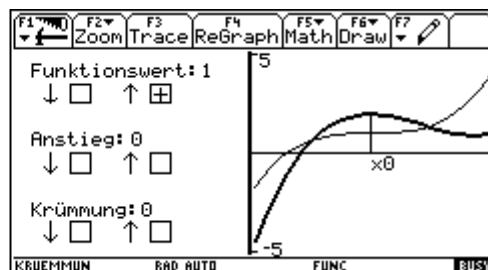
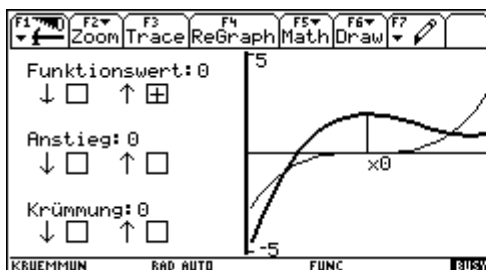
Bei den Schülern hat es sich bewährt, zunächst den Funktionswert anzupassen. Wir machen absichtlich einmal einen Fehler und verkleinern den Funktionswert.



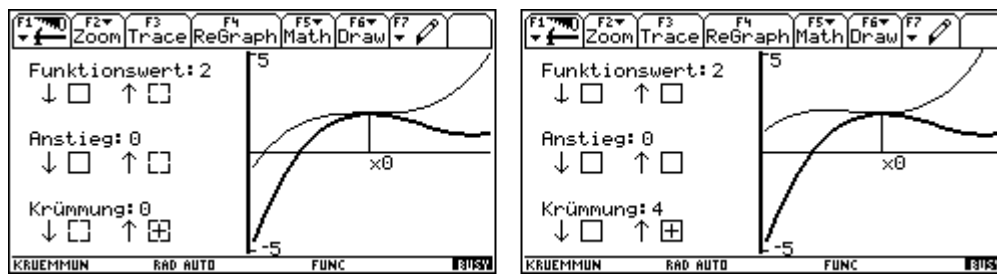
Nun müssen wir den Fehler verbessern.



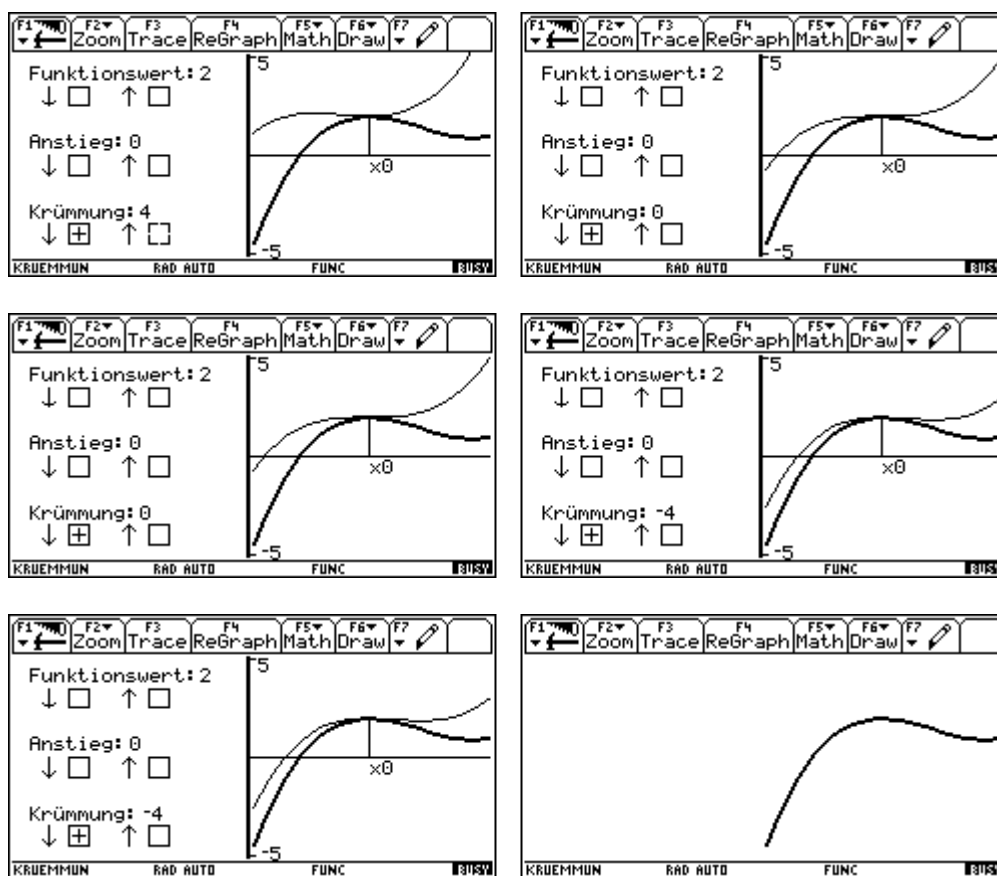
Und dann die richtige Anpassung vornehmen.



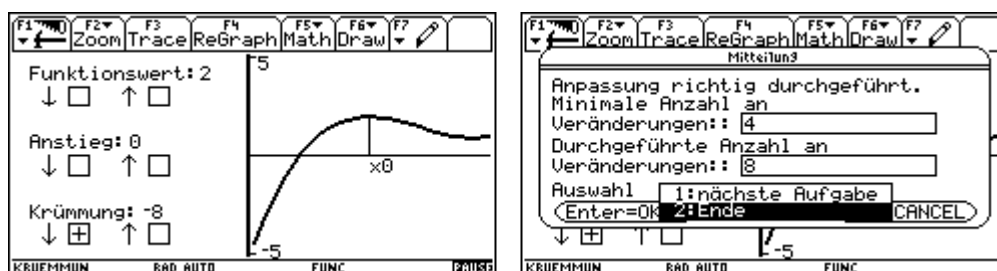
Der Anstieg muß nicht mehr angepaßt werden. Auch bei der Krümmung machen wir zunächst einen Fehler.



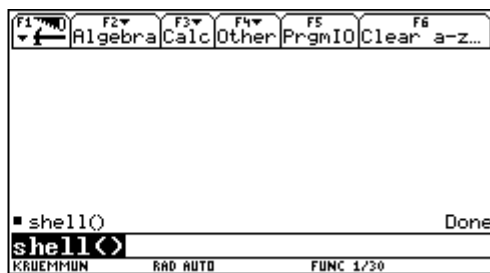
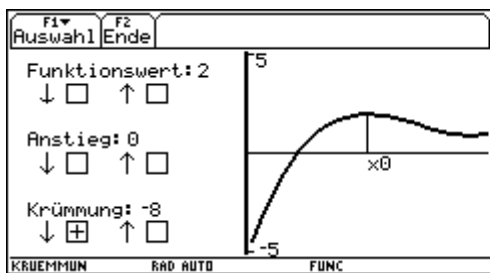
Dann passen wir die Krümmung schrittweise an, bis wir deckungsgleiche Graphen erreichen.



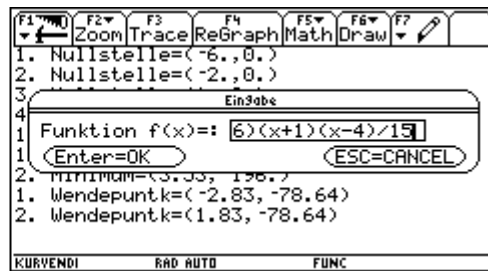
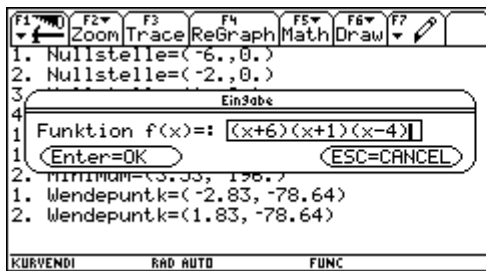
Mit **ENTER** sehen wir in einer Dialogbox eine Übersicht über unsere Bemühungen. Außerdem können wir wählen, ob wir noch eine Aufgabe bearbeiten oder beenden.



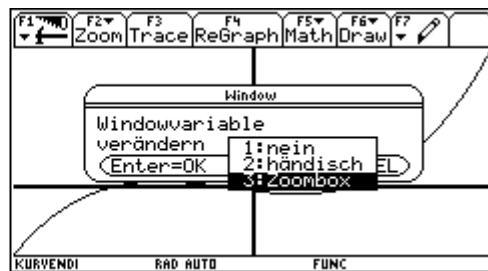
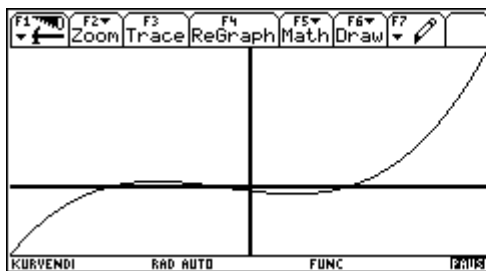
Mit **[F2]** verlassen wir die Programmgruppe endgültig.



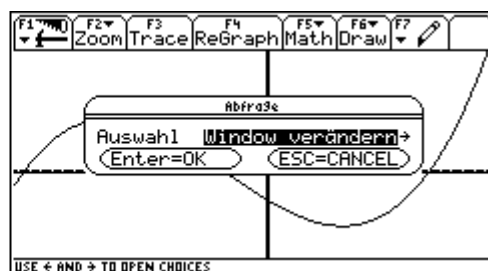
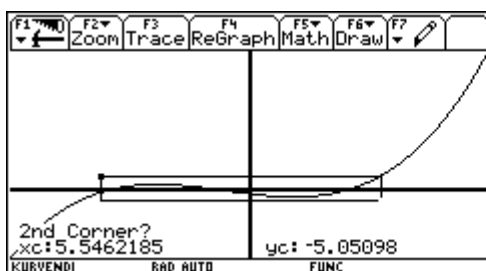
Wir geben eine Polynomfunktion ein.



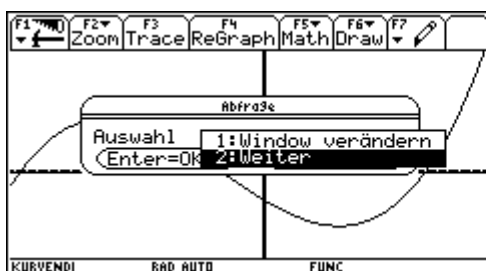
Zunächst wird der Graph im Bereich ZoomStd gezeichnet. Es folgt die Möglichkeit die Windowvariablen zu verändern.



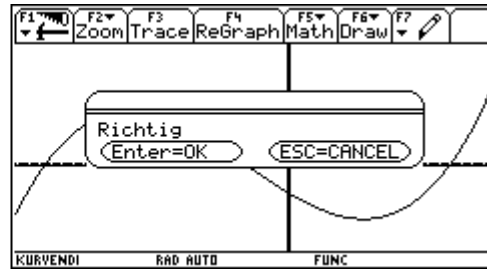
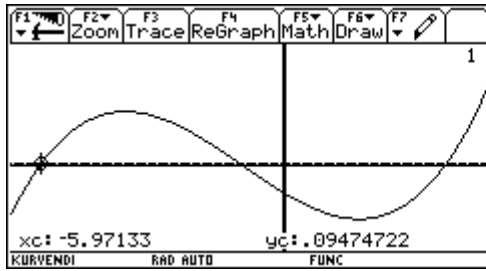
Wir wählen Zoombox und danach wählen wir keine weiter Veränderung der Windowvariablen.



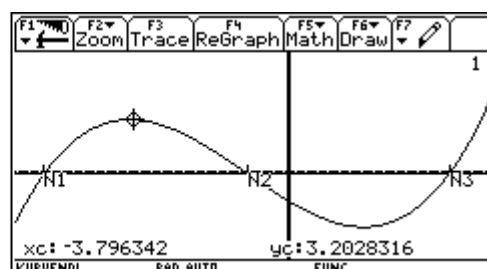
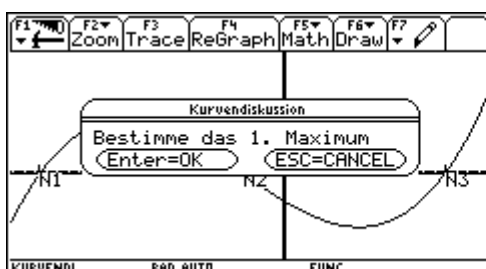
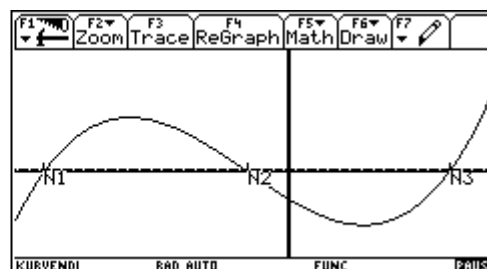
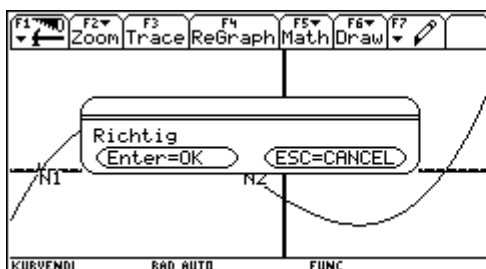
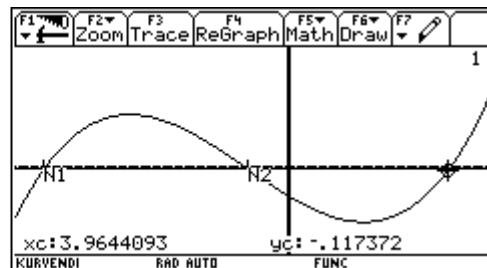
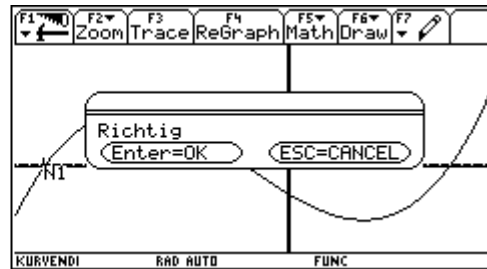
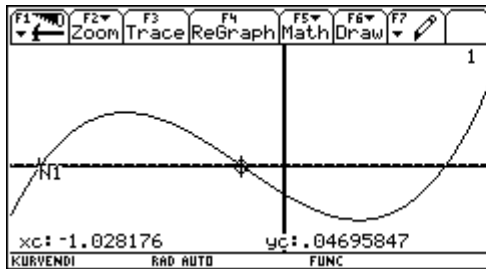
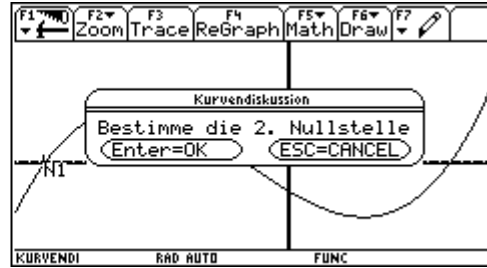
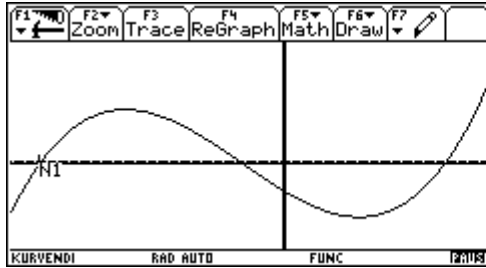
Nun werden wir aufgefordert mit dem im TRACE-Zustand befindlichen Graphikcursor die 1. Nullstelle (mit erster Nullstelle ist die kleinste gemeint) anzufahren.

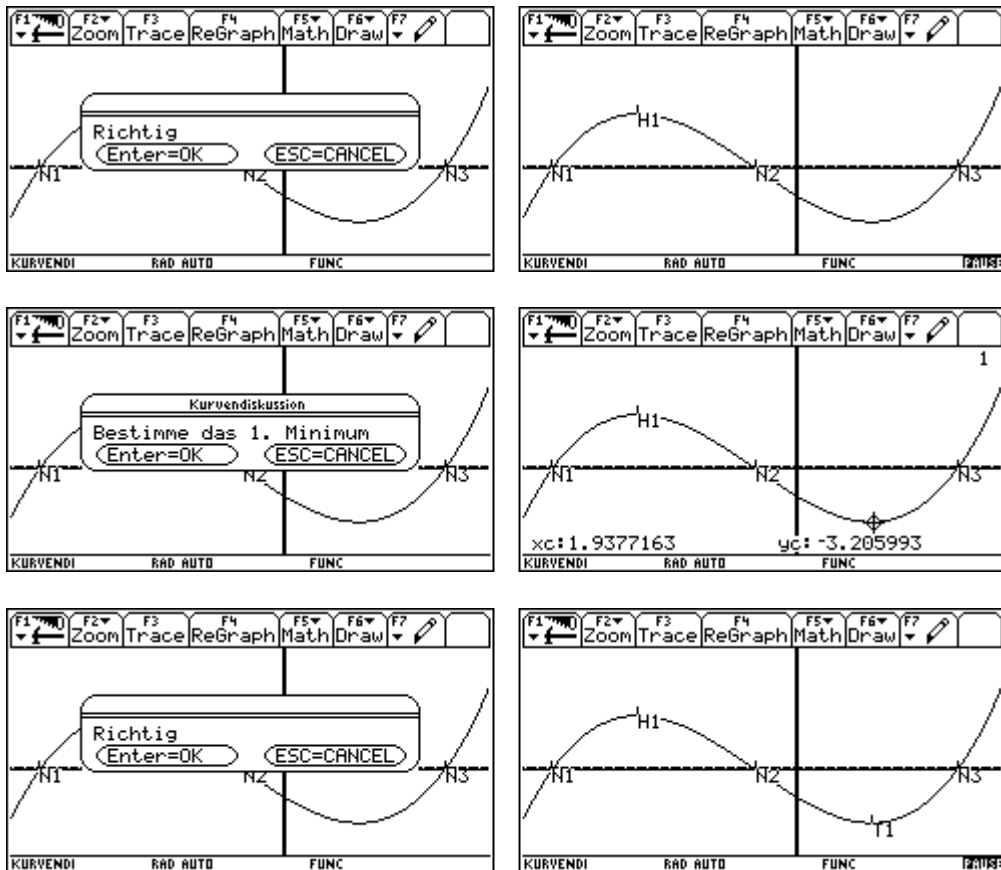


Wenn wir uns an der richtigen Stelle zu befinden glauben, bestätigen wir mit $\langle \text{ENTER} \rangle$. Das Programm akzeptiert einen kleinen Fehlbereich (+ -0,2) um die richtige Stelle. Daher ist das Programm für Funktionen, deren Nullstellen, Extremwert dichter liegen nicht sinnvoll. Waren wir innerhalb dieses Fehlbereichs, erhalten wir die Antwort richtig und die Stelle wird exakt markiert und beschriftet.

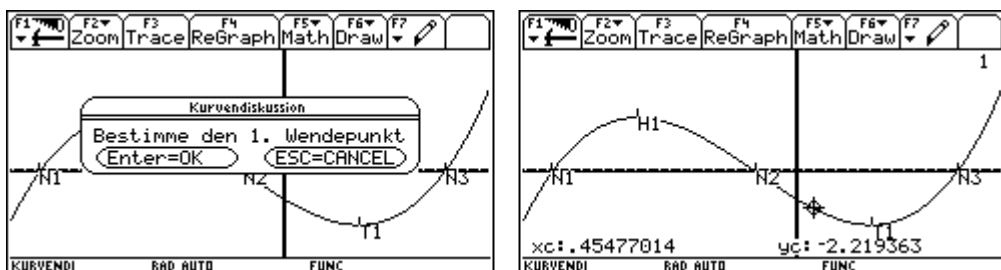


Es folgt die 2. Nullstelle und so fort.

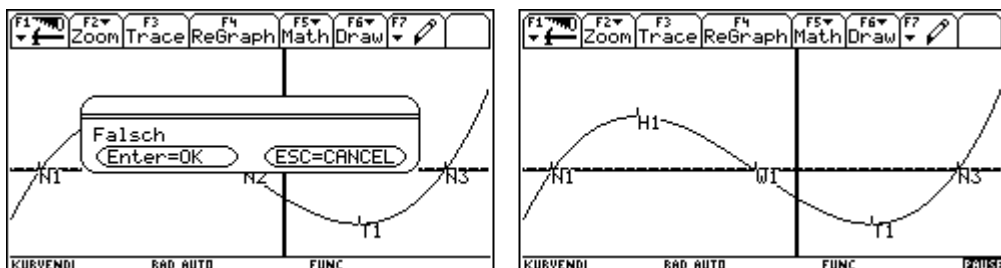




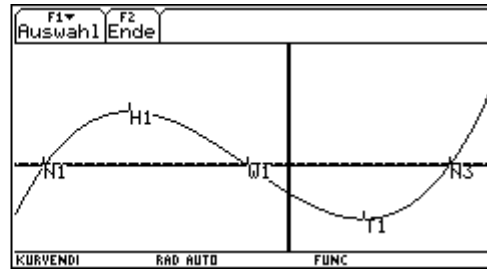
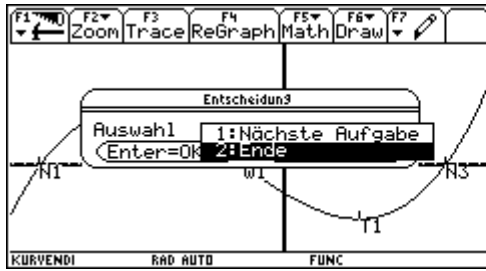
Beim Wendepunkt machen wir absichtlich einmal einen Fehler.



Das Programm antwortet mit falsch und beschriftet die richtige Stelle.

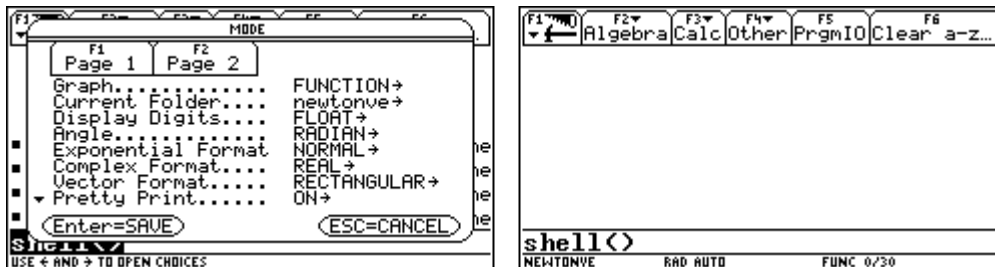


Am Ende gibt es die Wahlmöglichkeit zwischen einem weiteren Beispiel und dem Beenden des Programmteiles. Mit [F2] verlassen wir die Programmgruppe entgültig.

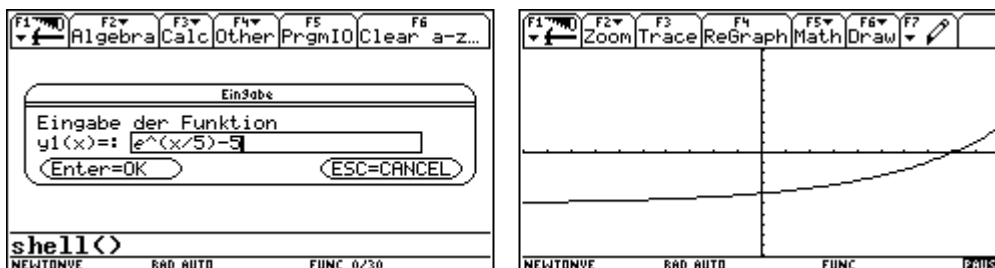


11. Programmgruppe: newtonve

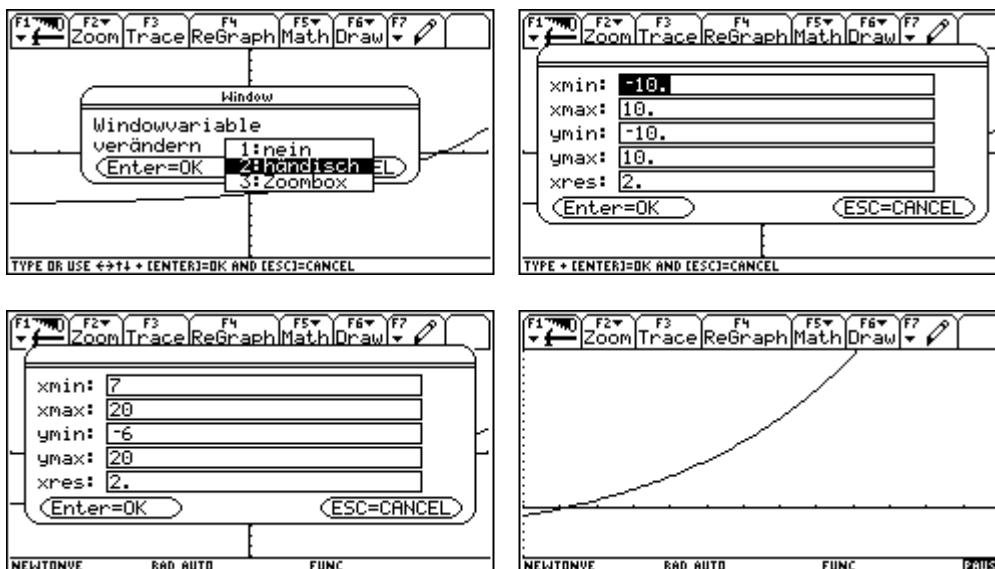
In den Folder newtonve wechseln und dann das Programm shell() starten.



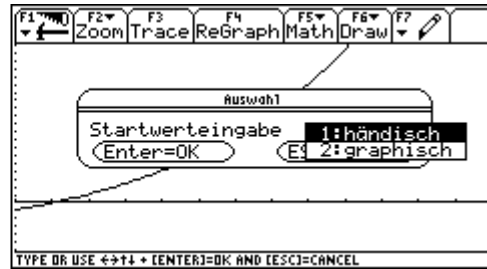
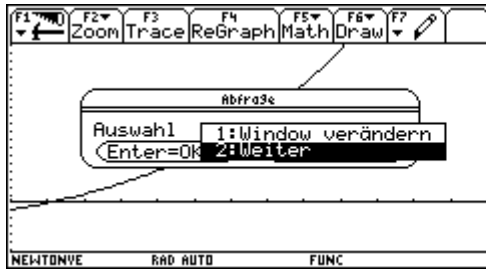
Dann die Funktion eingeben, für die das Verfahren gezeigt werden soll. Sie wird zunächst einmal im Bereich ZoomStd angezeigt.



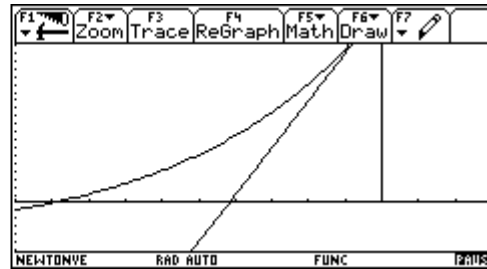
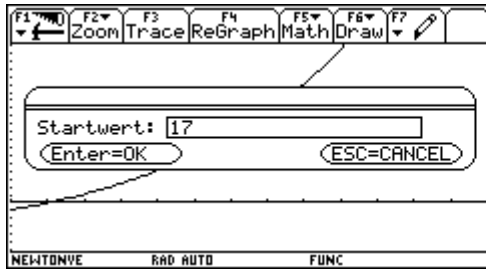
Nun können die Windowvariablen entsprechend verändert werden, um ein entsprechend günstiges Bild zu erhalten.



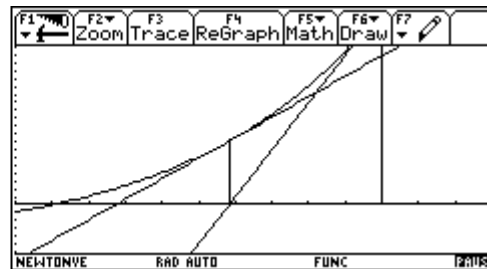
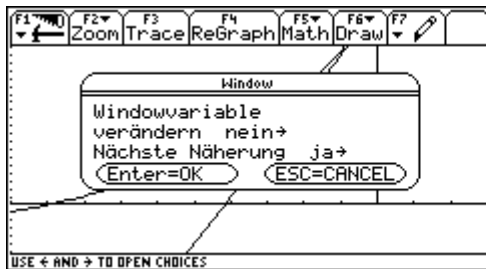
Wenn keine weitere Veränderung der Windowvariablen erwünscht wird, kann der Startwert des Verfahrens eingegeben werden.



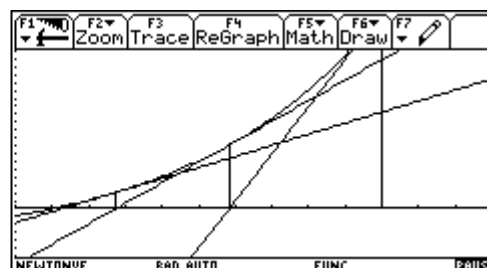
Wir wählen händisch und geben den Startwert exakt ein. Die Position des Startwertes wird durch eine normale zur x-Achse dargestellt und die Tangente wird eingezeichnet.



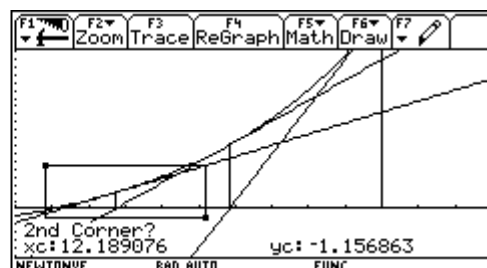
Wir haben die Möglichkeit, die Windowvariablen zu verändern oder die nächste Näherung anzuzeigen.



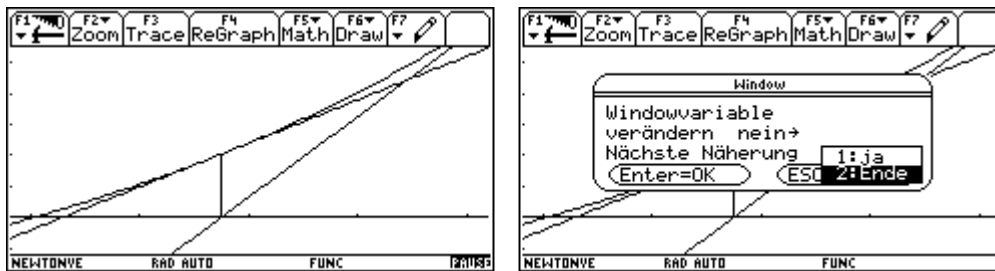
Auch diesmal wählen wir die nächste Näherung.



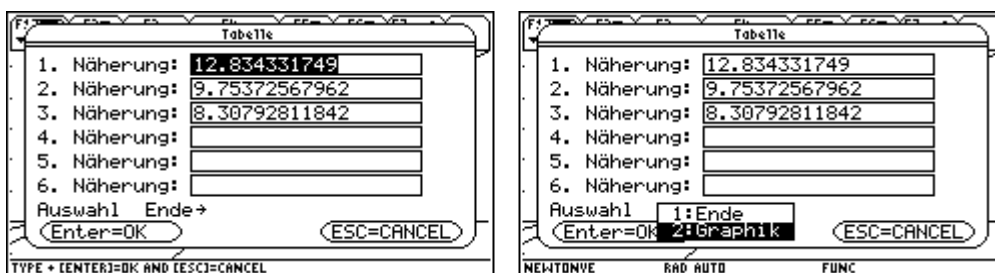
Nun wir die Darstellung zu unübersichtlich und wir wählen Zoombox und verkleinern den Ausschnitt um die Nullstelle.



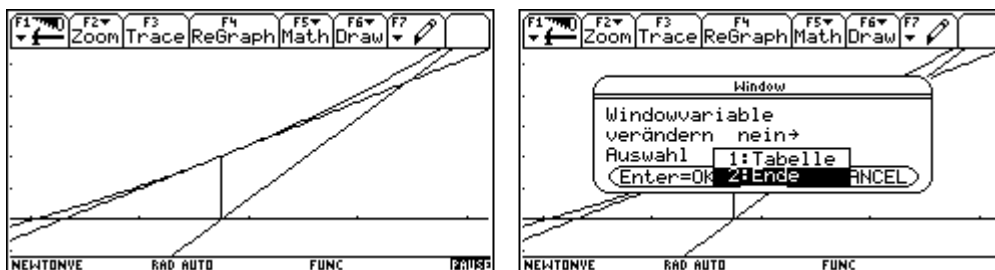
Dann wählen wir Ende.



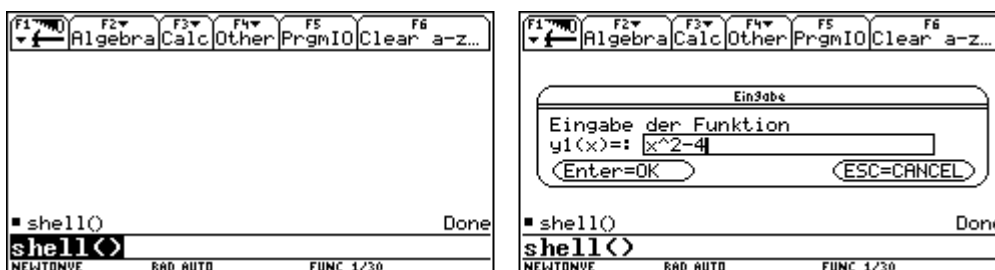
Eine Tabelle mit den Werten für die drei gezeichneten Näherungen erscheint. Wir sehen, dass das Programm maximal das Zeichnen von 6 Näherungen erlaubt. Wir können nun beenden oder zur Graphik zurückkehren.



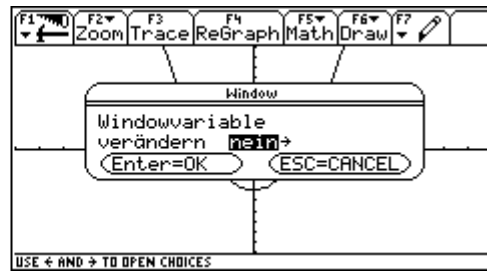
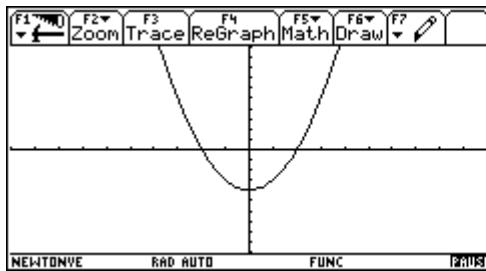
Zurückgekehrt in die Graphik können wir noch die Windovariablen verändern, zur Tabelle zurückkehren oder aufhören.



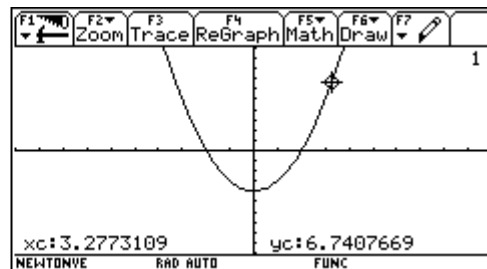
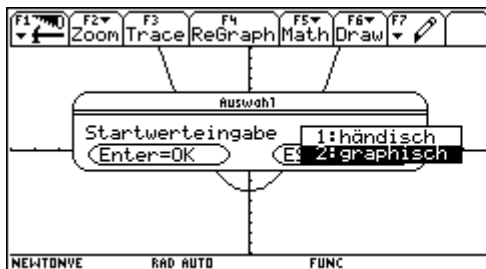
Nun wollen wir den Startwert graphisch festlegen. Wir starten neu und geben eine Parabel ein.



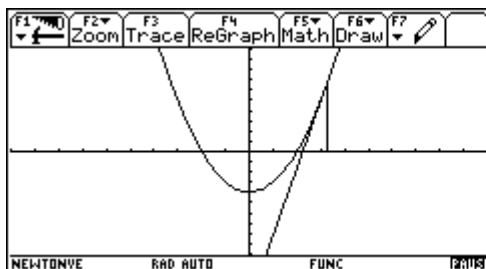
Dann überspringen wir den nächsten Menüpunkt.



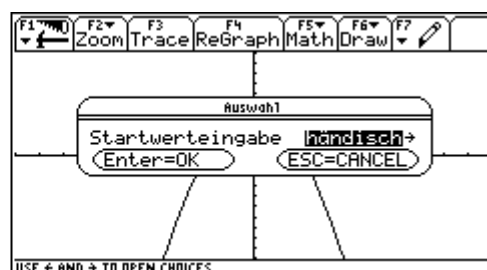
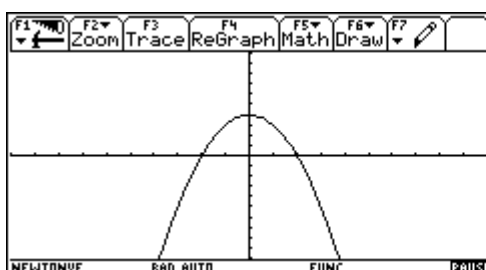
Dann wählen wir graphisch. Der Cursor sitzt im TRACE-Zustand am Graphen. Wir bewegen ihn zur gewünschten Stelle und bestätigen mit **[ENTER]**.



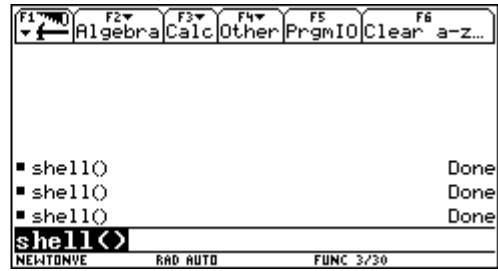
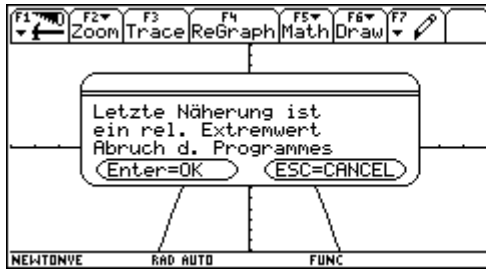
Die gewählte Stelle wird als Startwert verwendet. Das Programm ist dagegen gesichert, dass sich unter dem Startwert oder den Näherungswerten ein relatives Extremum befindet. Wir starten neu und geben die folgende Parabel ein.



Wir überspringen nun die Fenstereinstellung und wählen die händische Startwerteinstellung.



Wir geben als Startwert 0 ein und erhalten die folgende Antwort.

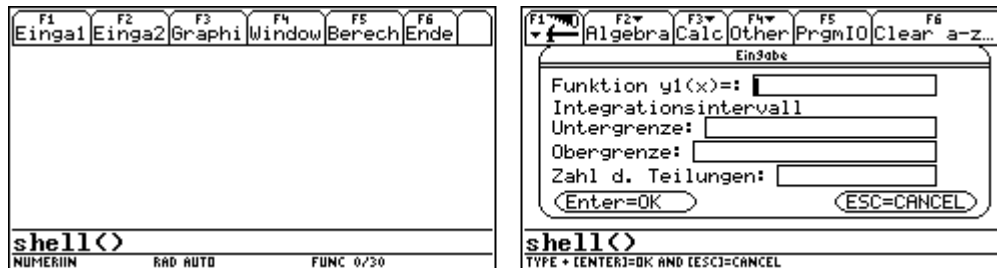


12. Programmgruppe: numeriin

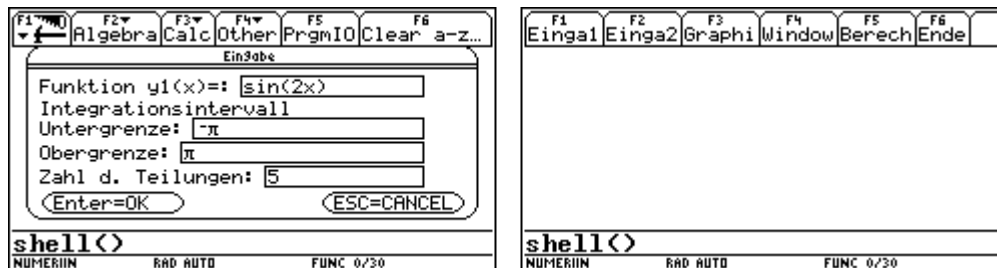
In den Folder numeriin wechseln und das Programm shell() starten.



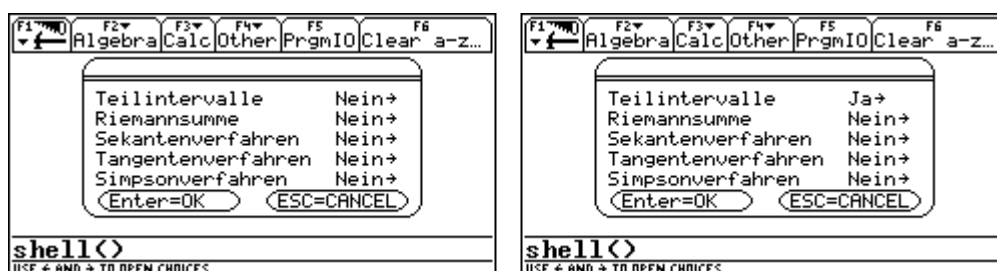
Es erscheint folgende Menüleiste. Wir wählen [F1] und erhalten die folgende Eingabebox. Die hier eingestellten Werte beziehen sich auf die graphische Darstellung der Verfahren und die Berechnung. Können aber später jederzeit noch verändert werden.



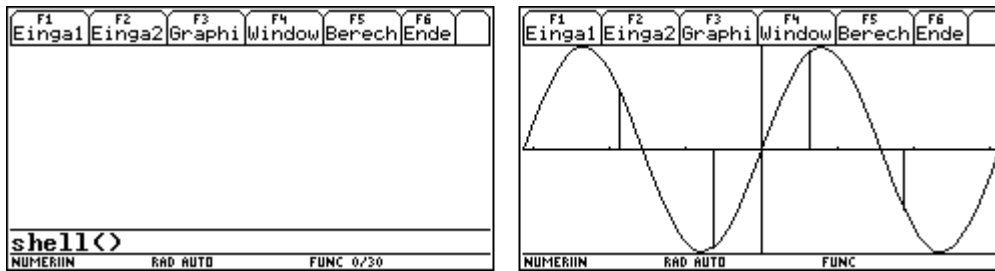
Wir wählen folgende Einstellung.



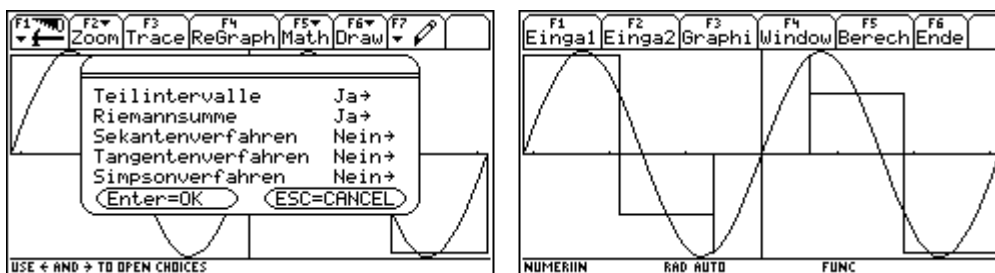
Mit [F2] gelangen wir zu folgender Box. Hier können die Verfahren ausgesucht werden, die tatsächlich graphisch dargestellt werden sollen. Das Simpsonverfahren wird nur bis zu einer Teilungszahl 10 graphisch dargestellt. Wir wählen zunächst nur die Aufteilung des Intervalles.



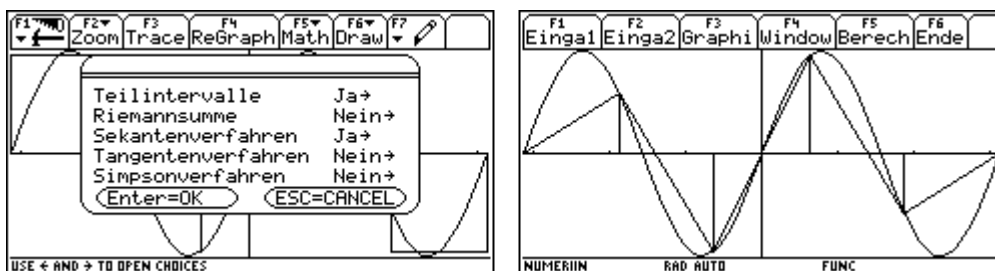
Man kehrt zur Menüleiste zurück. Immer wenn bei [F1] oder [F2] Veränderungen vorgenommen wurden, muß die Graphik über [F3] und die Berechnung über [F5] aktualisiert werden. Wir wählen [F3] und erhalten folgende Graphik.



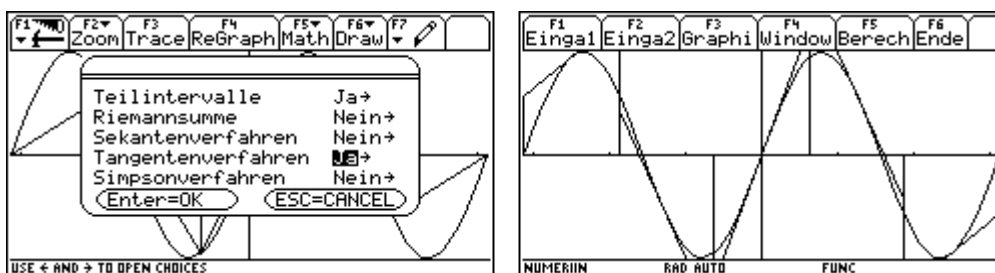
Bei Riemannsumme ist der Funktionswert in der Mitte des Teilintervalles gewählt.



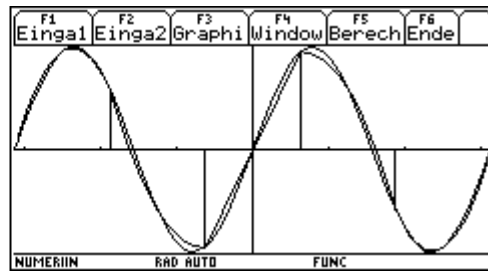
Das Sekantenverfahren verwendet die Sehnen über den Teilintervallen.



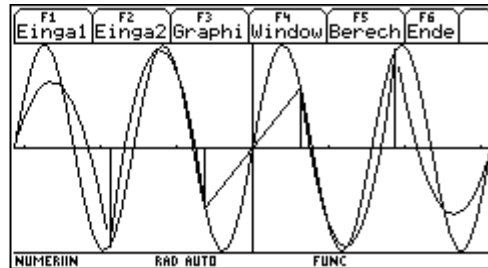
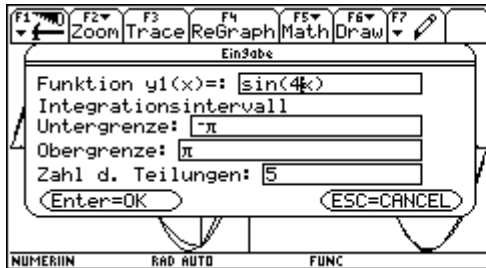
Das Tangentenverfahren benutzt die Tangenten über den Teilintervallen.



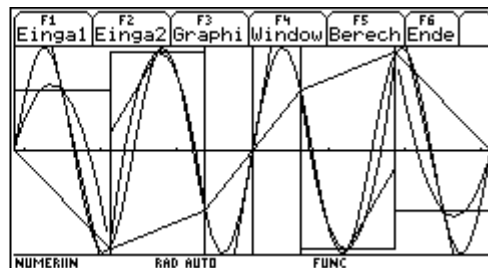
Das Simpsonverfahren benutzt Parabeln.



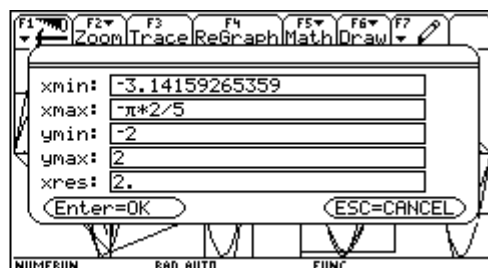
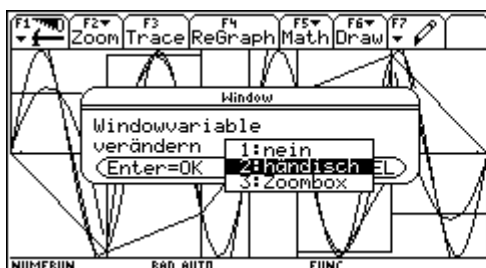
Um die Parabeln besser zu zeigen verändern wir in [F1] die Funktion und aktualisieren über [F3].



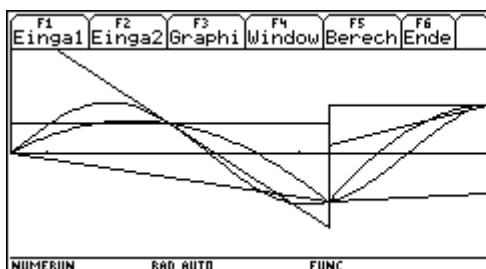
Nun wählen wir in [F2] alle Verfahren und aktualisieren mit [F3]



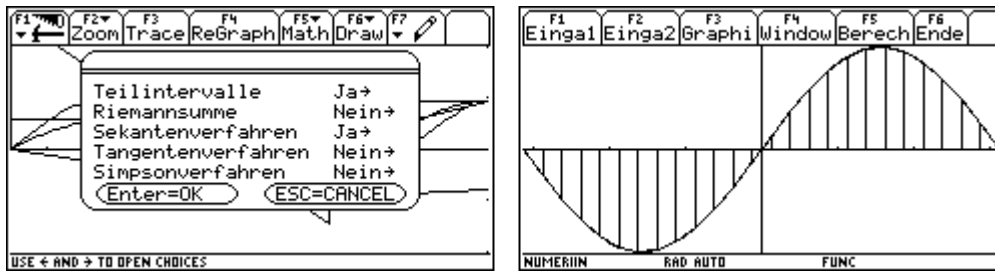
Über [F4] können die Windowvariablen verändert werden. Wir machen davon Gebrauch um alle Verfahren in einem Teilintervall bewundern zu können.



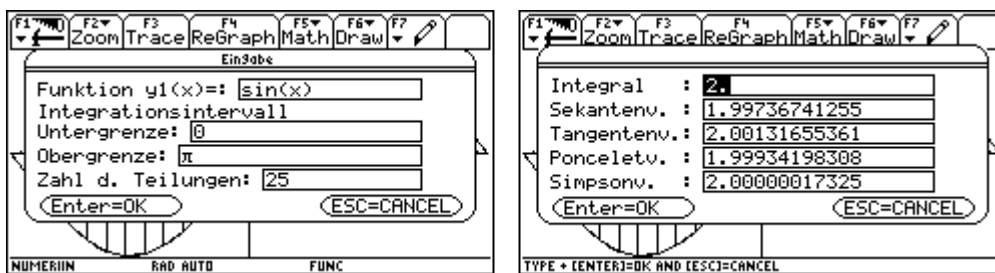
Dann verändern wir die Angaben in [F1], um größere Teilungszahlen darstellen zu können.



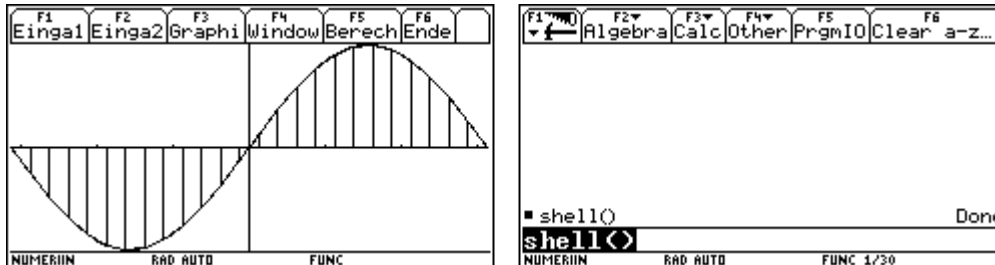
Das Simpsonverfahren dürfen wir jetzt nicht mehr wählen. Aber für das Sekantenverfahren sieht man für diese Funktion schon eine sehr gute Näherung.



Für die Berechnung verändern wir noch einmal unsere Angaben und erhalten mit **[F5]** eine Tabelle mit den verschiedenen Verfahren. Das Poncelet-Verfahren ist das arithmetische Mittel von Sekantenverfahren und Tangentenverfahren.



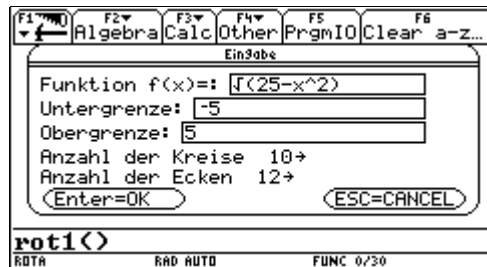
Mit **[F6]** wird diese Programmgruppe verlassen.



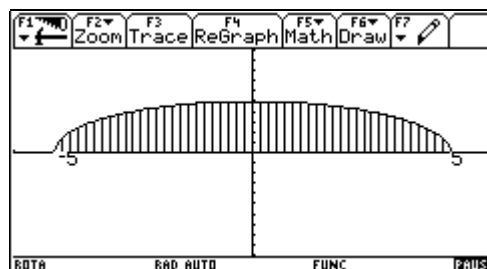
13. Programmgruppe: rotation

In den Folder rotation wechseln und das Programm shell() starten.

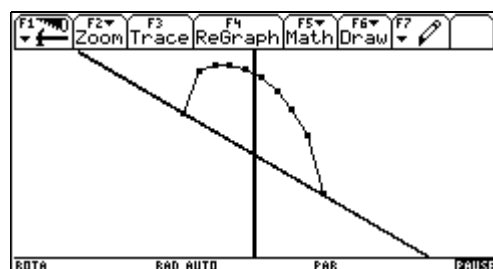
In der folgenden Dialogbox muß die Funktion eingegeben werden, deren Graph rotieren soll, ferner die Ober- und Untergrenze des Intervalles, um das Stück des Graphen festzulegen, das rotieren soll. Dann müssen für das Drahtgittermodell, durch das der Rotationskörper dargestellt wird, die Anzahl der Ecken der Vielecke, die die Kreise nähern, und die Anzahl dieser genäherten Kreise eingegeben werden.



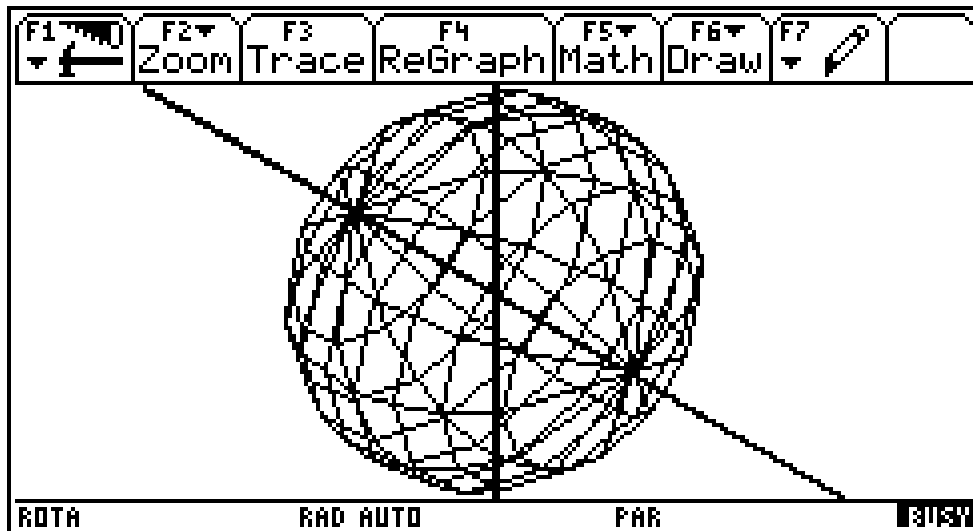
Der Graph im festgelegten Intervall wird dargestellt. Die Fläche zwischen Graphen und x-Achse ist schraffiert.



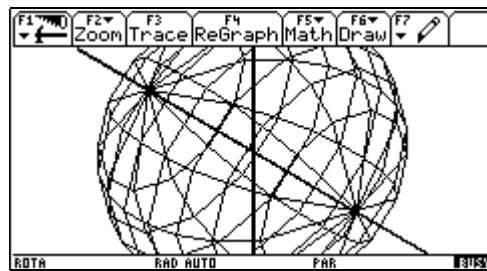
Das nächste Bild zeigt die x-Achse, die y-Achse und den Graphen im festgelegten Intervall in einer axiometrischen Darstellung.



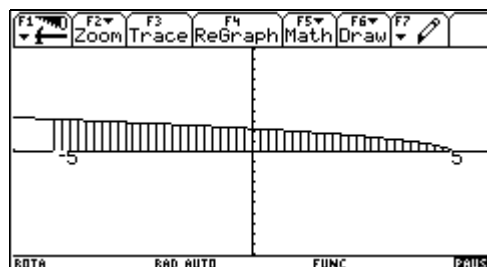
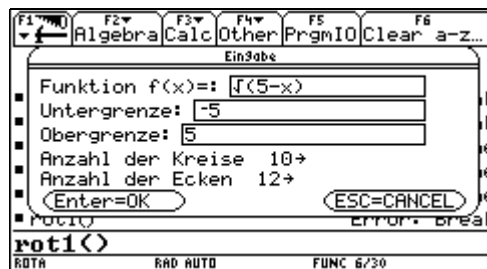
Nun werden die Berechnung für die Darstellung des Rotationskörpers gemacht. Diese Berechnungen können bis zu 2 Minuten dauern. Dann erscheint nach Betätigen von **[ENTER]** der Rotationskörper.

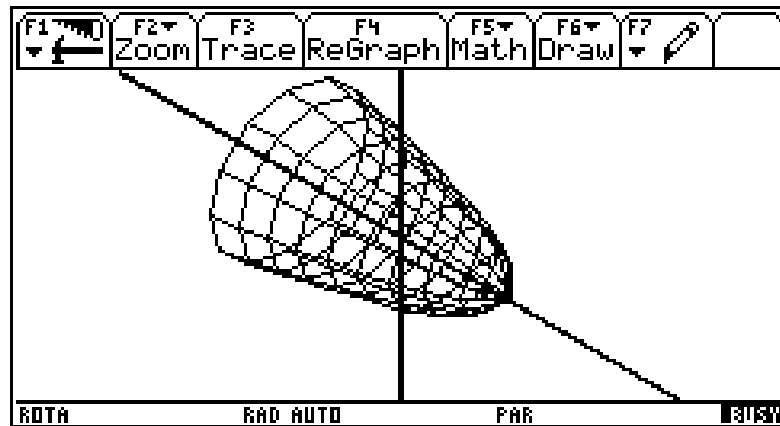


Mit der Taste k kann der Bildschirmausschnitt verkleinert werden. Mit der Taste g kann der Bildschirmausschnitt vergrößert werden.

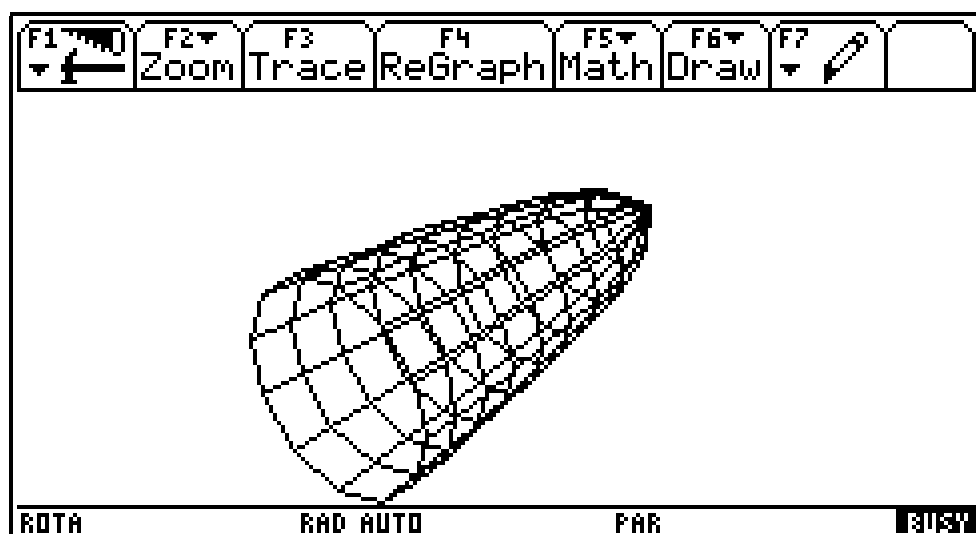
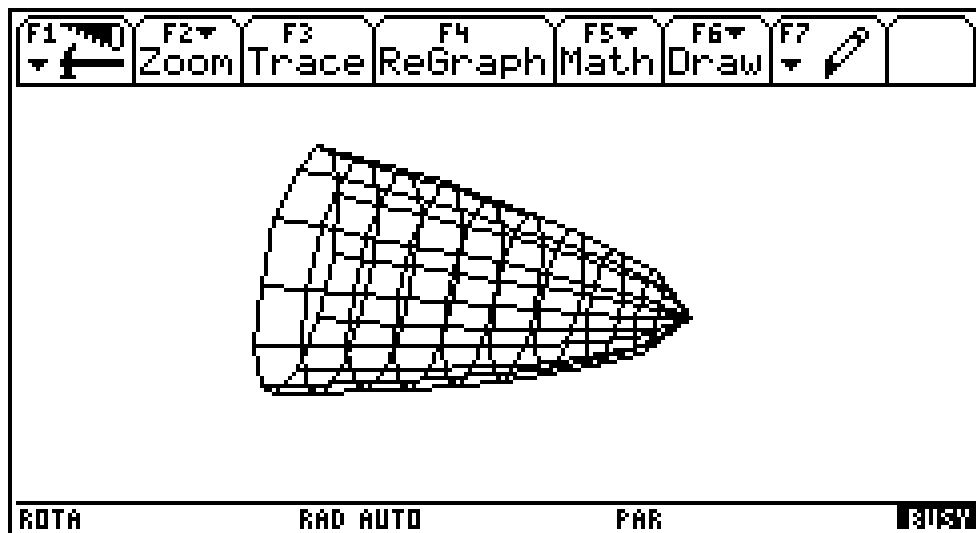


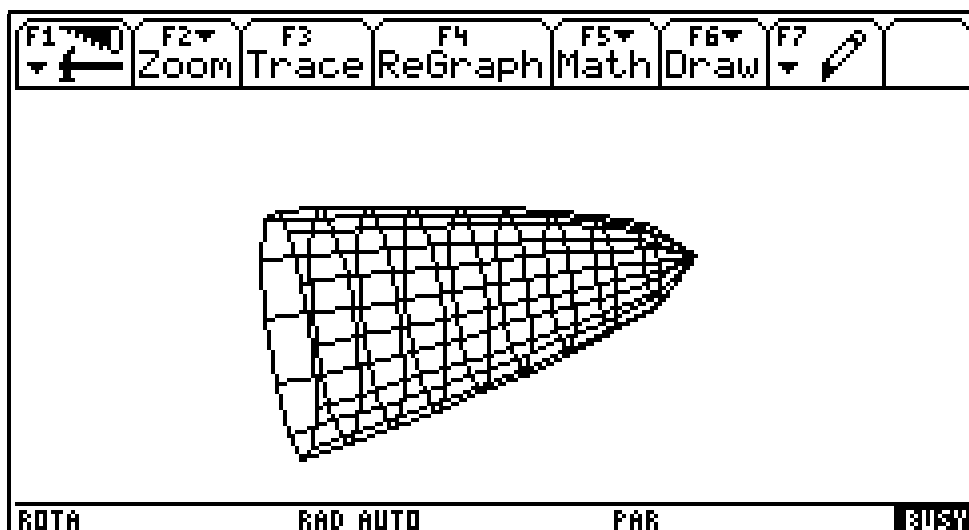
Als zweites Beispiel wollen wir den Rotationskörper einer Parabel zeigen.





Mit der Taste d kann der Drehkörper um die y-Achse stückweise rotieren, wobei von Darstellung zu Darstellung auch bis zu 2 Minuten verstreichen können.



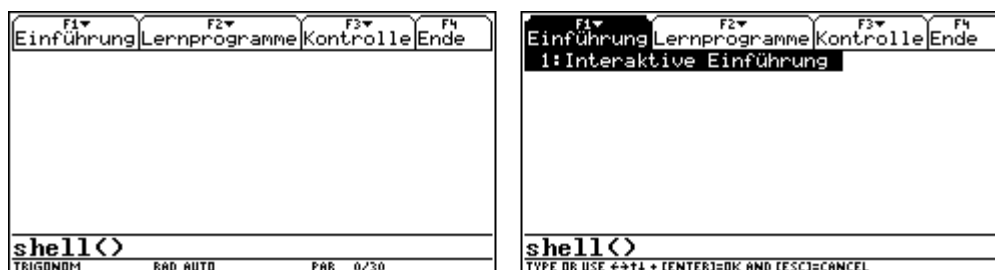


14. Programmgruppe: trigonom

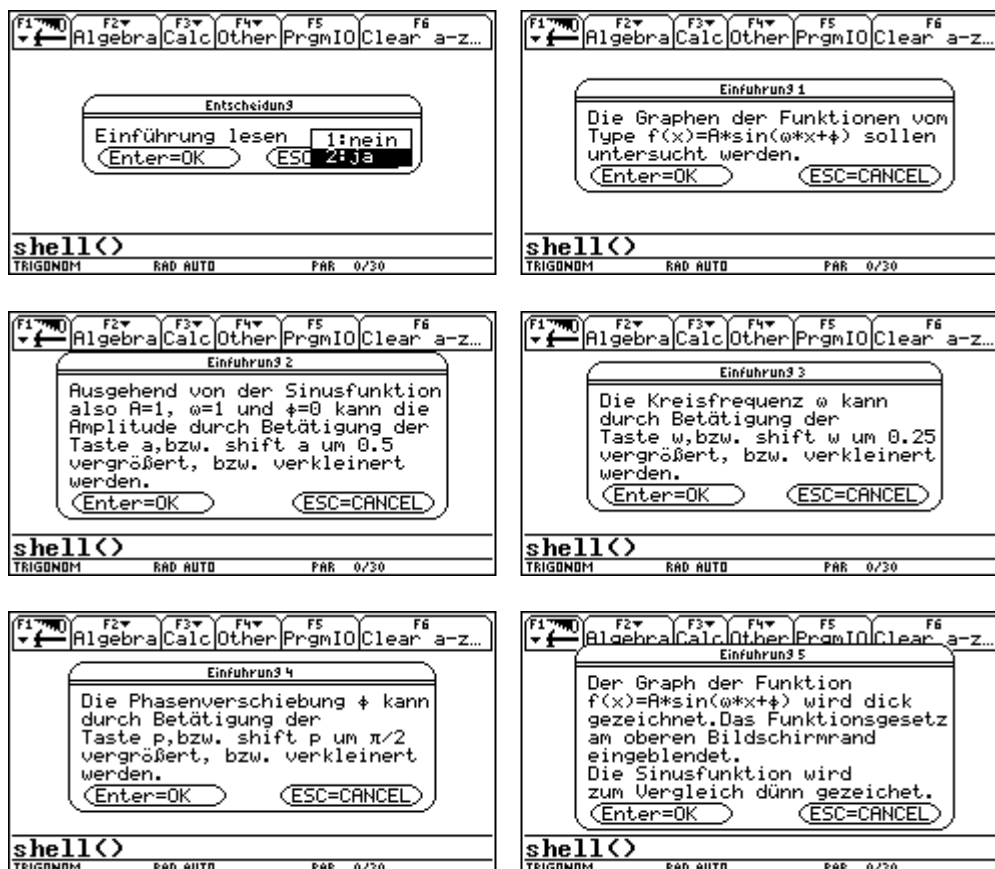
In den Folder trigonom wechseln und das Programm shell() starten.



Bei **F1** die interaktive Einführung wählen.

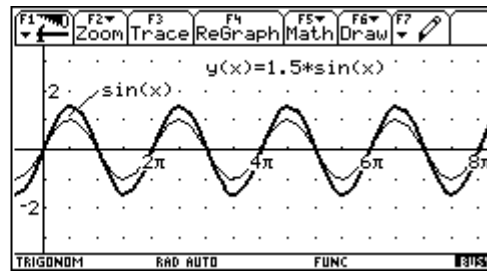
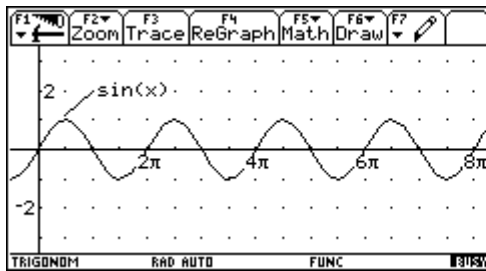


Hier gibt es die Möglichkeit eine Einführung zu lesen.

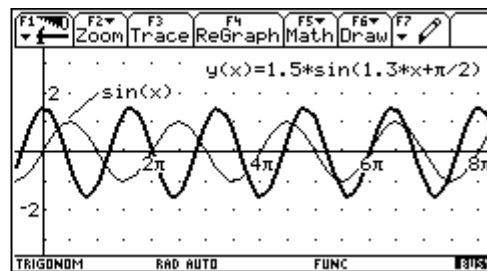
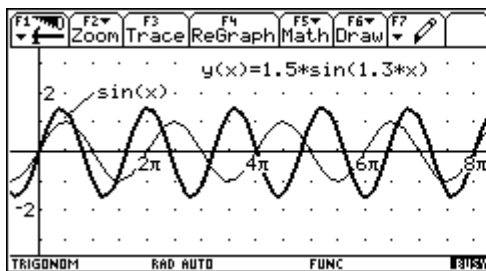




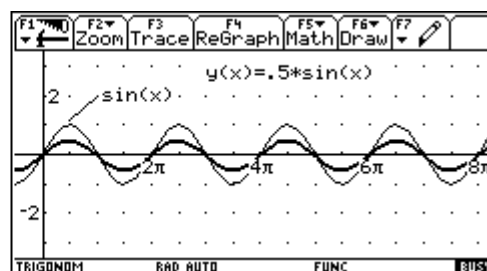
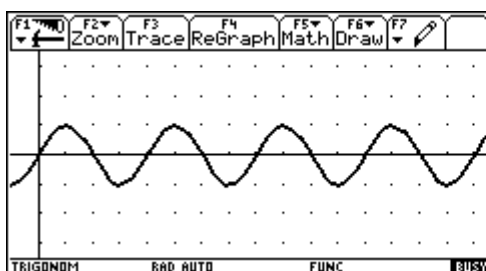
Nach dem Lesen der Einführung startet sofort das Programm. Mit der Taste a wird die Amplitude vergrößert.



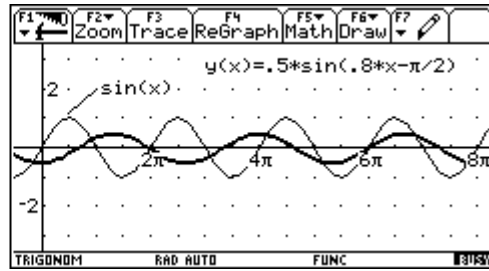
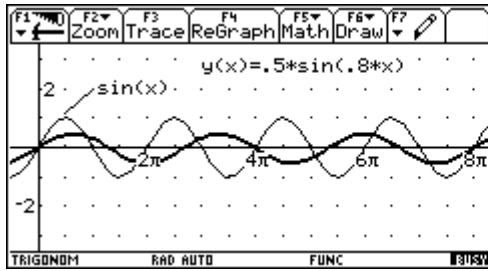
Mit der Taste w wird die Kreisfrequenz vergrößert und mit der Taste p wird die Phasenverschiebung vergrößert.



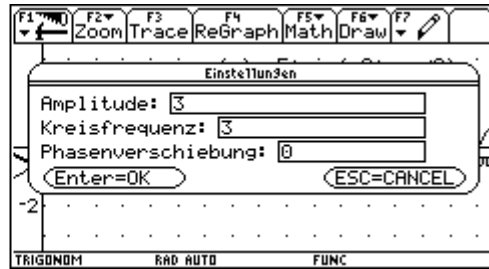
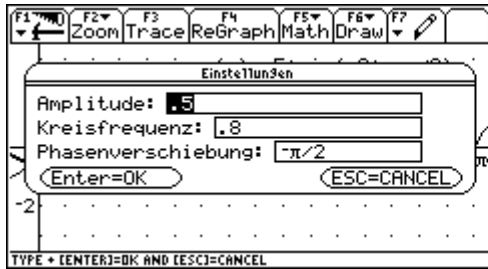
Mit der Taste z stellen wir den Anfangszustand wieder her. Mit \uparrow a wird die Amplitude verkleinert.



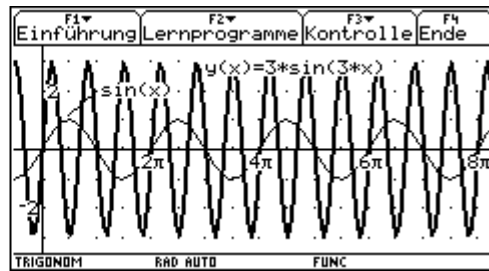
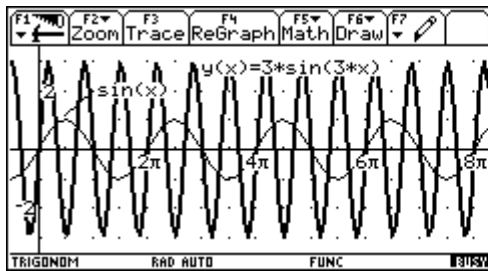
Mit \uparrow w wird die Kreisfrequenz verkleinert und mit \uparrow p wird die Phasenverschiebung verkleinert.



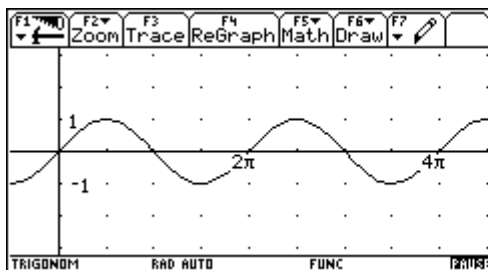
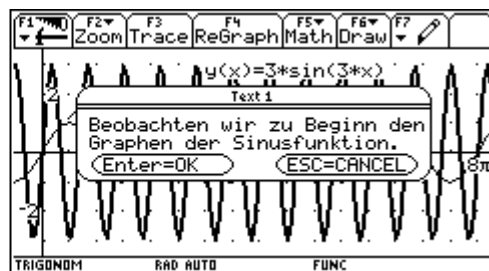
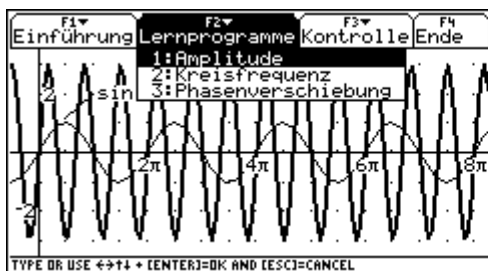
Mit der Taste e kann eine Dialogbox aufgerufen werden, in der alle Werte auf einmal umgestellt werden können.

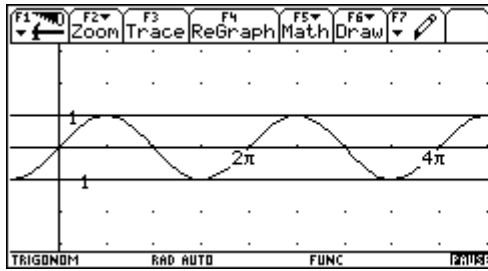


2^{nd} \boxed{ESC} wird dieser Programmteil verlassen.

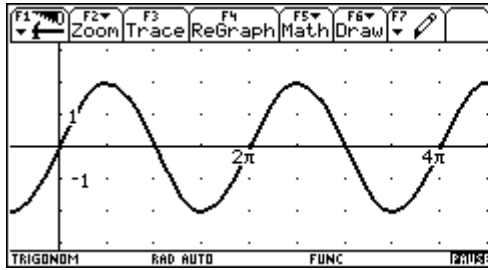


Wir wenden uns den Lernprogrammen unter $\boxed{F2}$ zu. Zunächst dem Lernprogramm über die Amplitude.

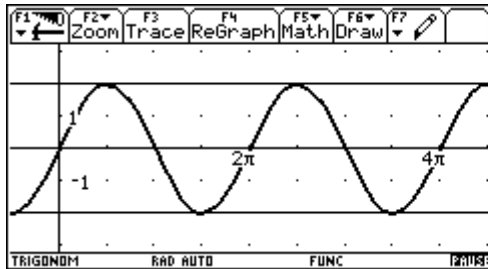




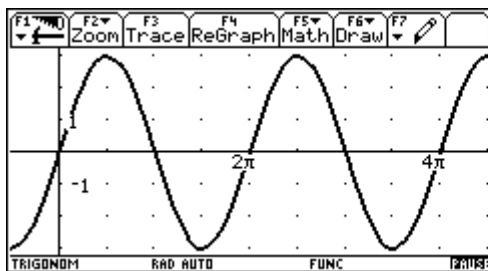
Text 3
 Beobachten wir nun den Graphen der Funktion $f(x)=2*\sin(x)$.
 (Enter=OK) (ESC=CANCEL)



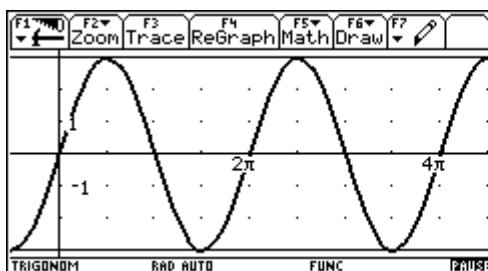
Text 4
 Die Maxima besitzen den Funktionswert 2, die Minima besitzen den Funktionswert -2.
 (Enter=OK) (ESC=CANCEL)



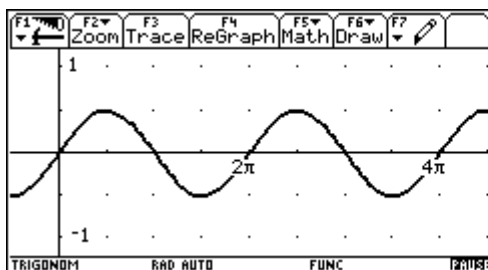
Text 5
 Beobachten wir nun den Graph der Funktion $f(x)=3*\sin(x)$.
 (Enter=OK) (ESC=CANCEL)



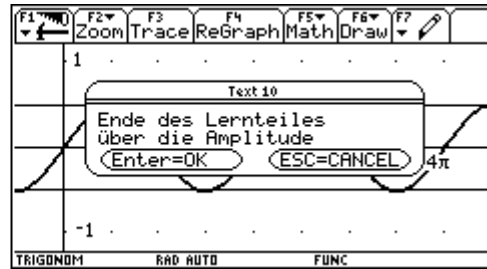
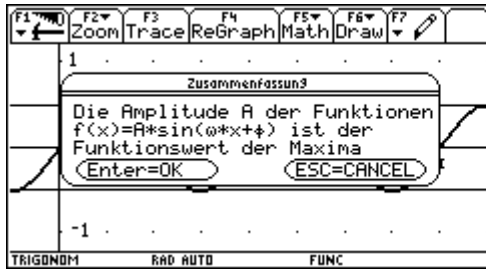
Text 6
 Die Maxima besitzen den Funktionswert 3, die Minima besitzen den Funktionswert -3.
 (Enter=OK) (ESC=CANCEL)



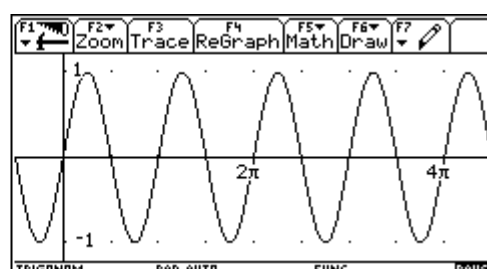
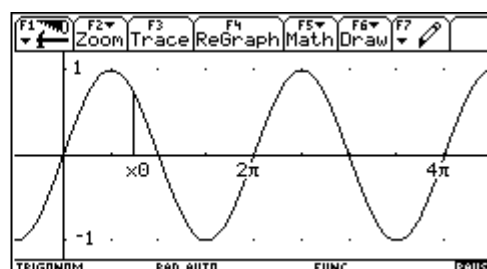
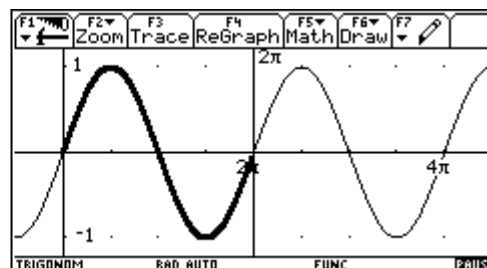
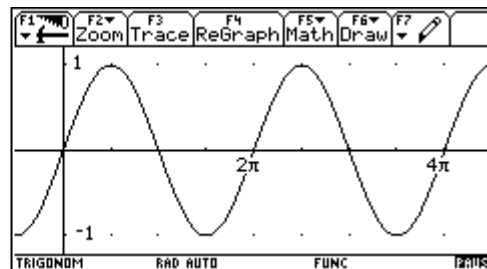
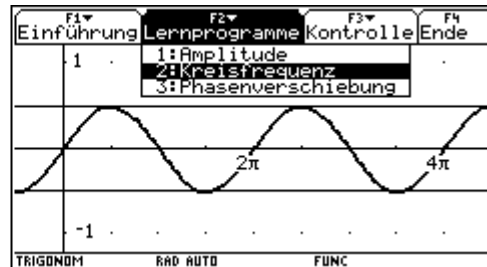
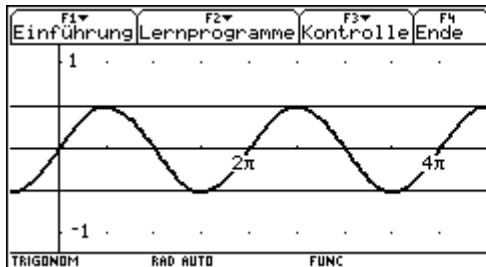
Text 7
 Beobachten wir nun den Graphen der Funktion $f(x)=1/2*\sin(x)$.
 (Enter=OK) (ESC=CANCEL)

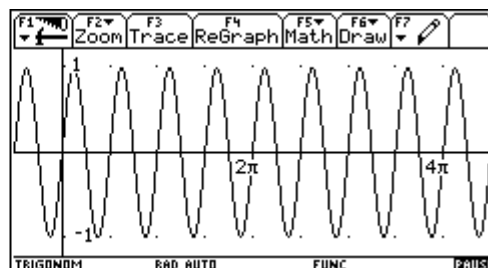
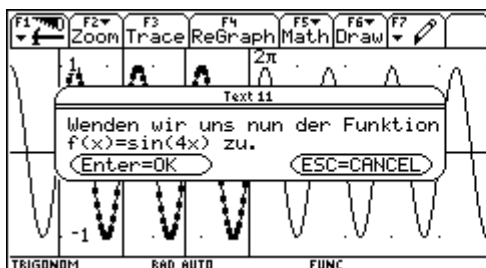
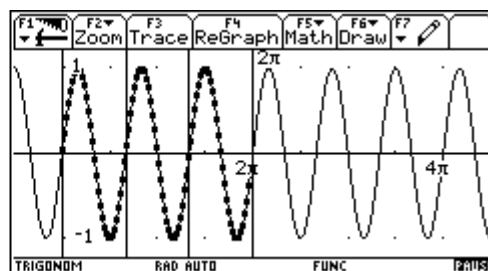
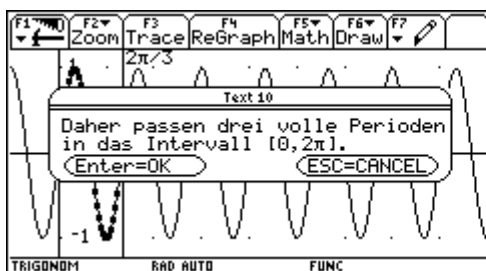
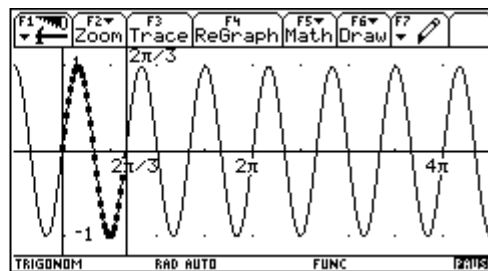
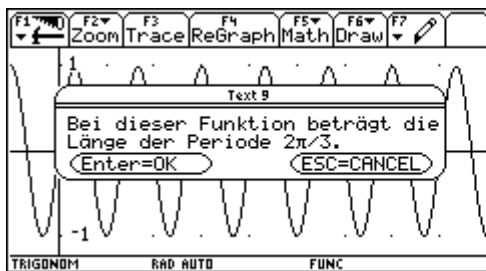
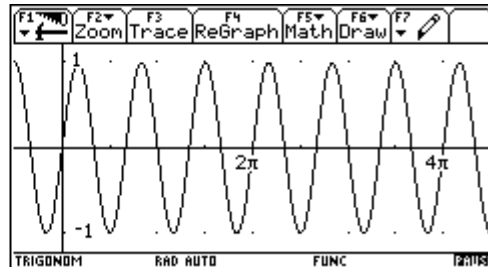
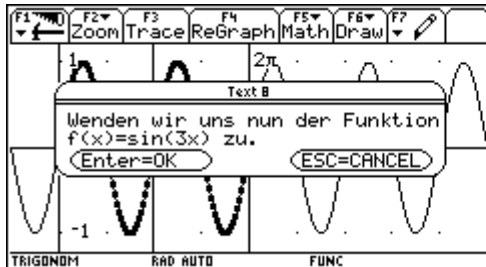
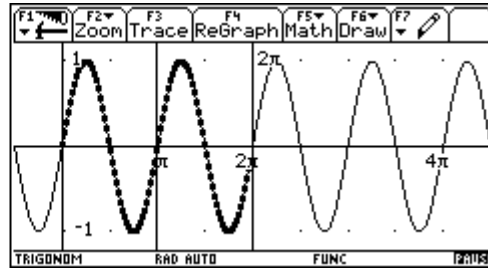
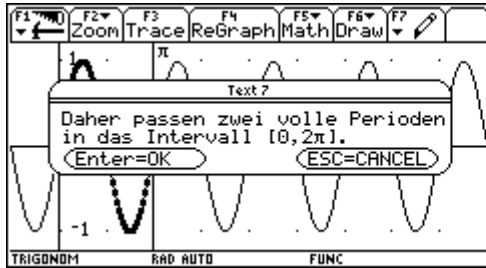
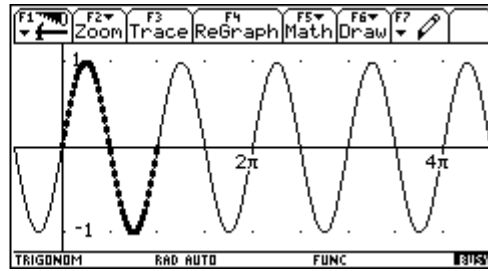
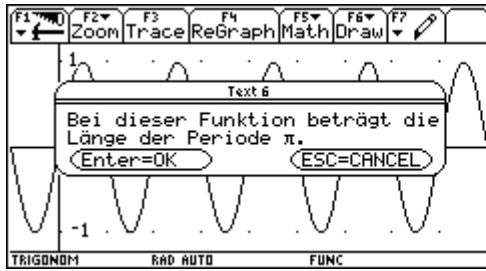


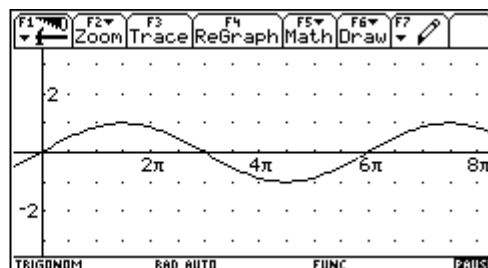
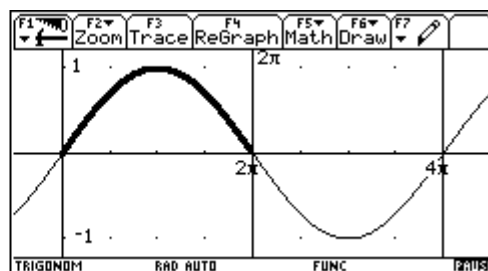
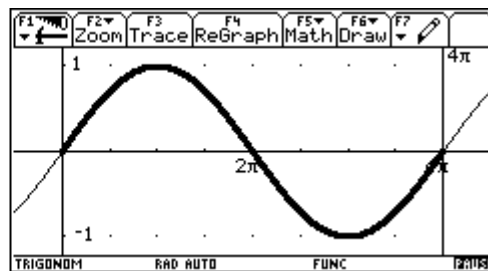
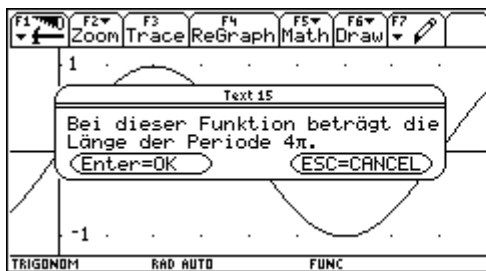
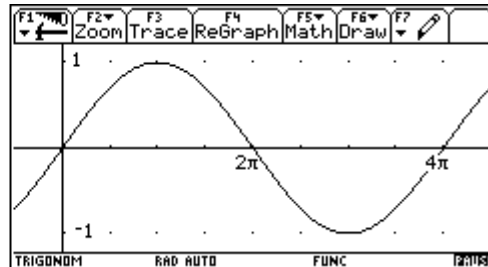
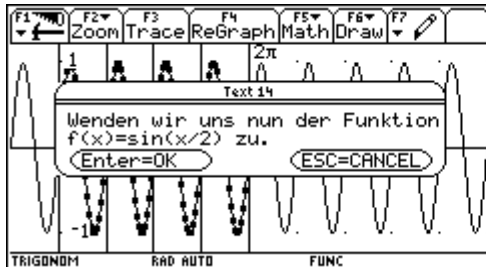
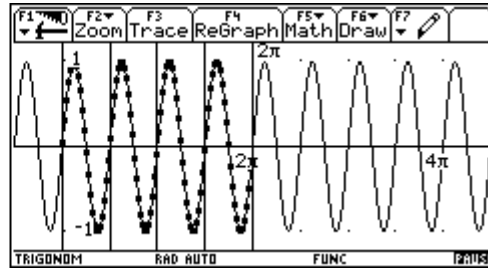
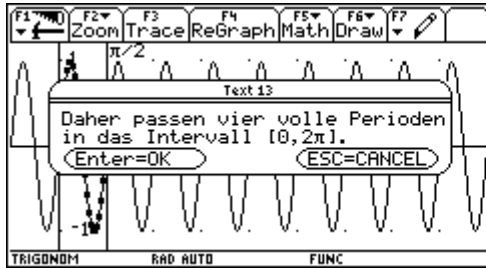
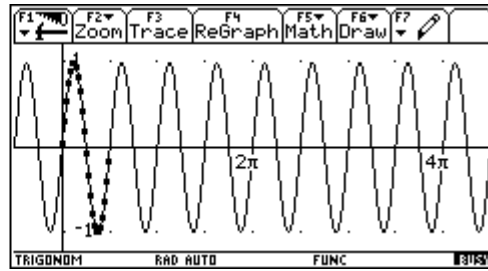
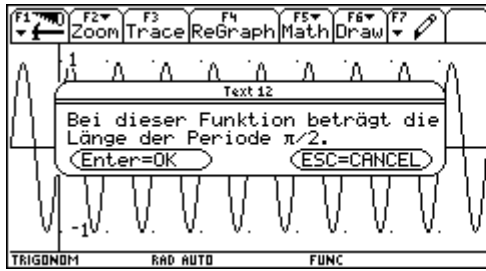
Text 8
 Die Maxima besitzen den Funktionswert 1/2, die Minima besitzen den Funktionswert -1/2.
 (Enter=OK) (ESC=CANCEL)

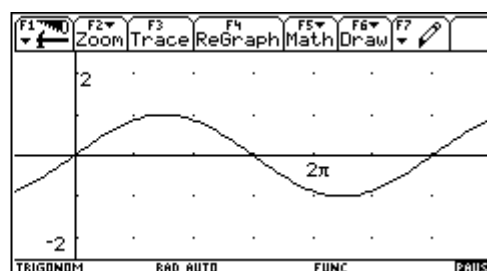
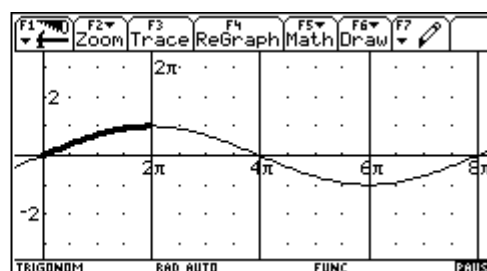
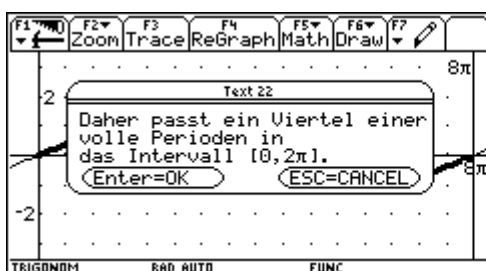
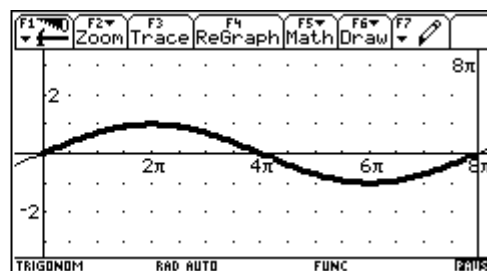
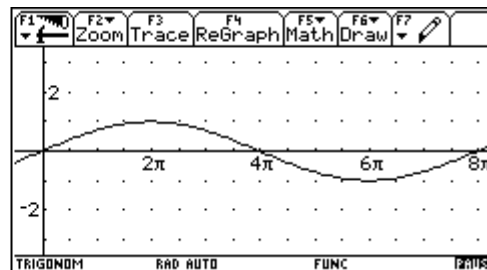
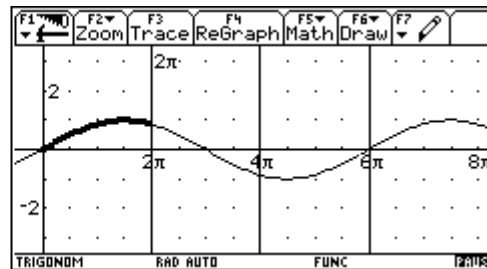
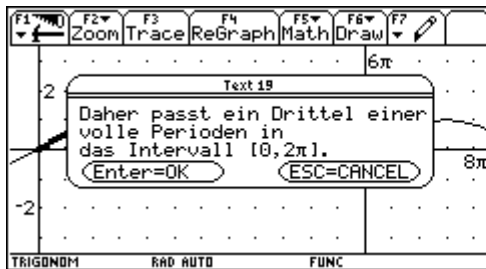
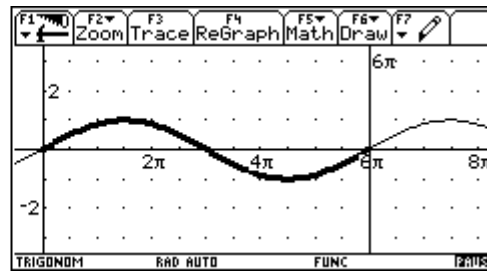
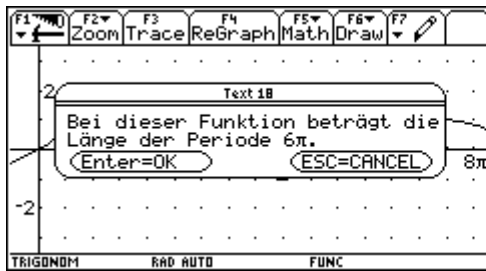


Dann betrachten wir den Lernteil über die Kreisfrequenz.









TRIGONDM RAD AUTO FUNC

F1 F2 F3 F4 F5 F6 F7

Zoom Trace ReGraph Math Draw

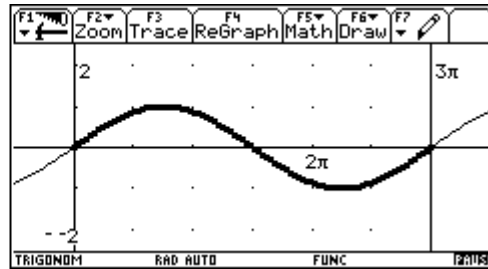
2

Text 24

Bei dieser Funktion beträgt die Länge der Periode 3π .

Enter=OK ESC=CANCEL

-2



TRIGONDM RAD AUTO FUNC

F1 F2 F3 F4 F5 F6 F7

Zoom Trace ReGraph Math Draw

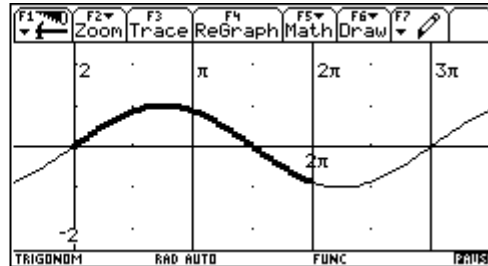
2

Text 25

Daher passt zwei Drittel einer vollen Periode in das Intervall $[0, 2\pi]$.

Enter=OK ESC=CANCEL

-2



TRIGONDM RAD AUTO FUNC

F1 F2 F3 F4 F5 F6 F7

Zoom Trace ReGraph Math Draw

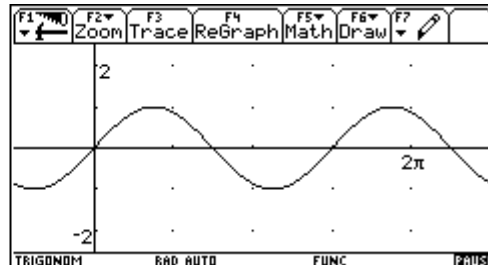
2

Text 26

Wenden wir uns nun der Funktion $f(x) = \sin(4x/3)$ zu.

Enter=OK ESC=CANCEL

-2



TRIGONDM RAD AUTO FUNC

F1 F2 F3 F4 F5 F6 F7

Zoom Trace ReGraph Math Draw

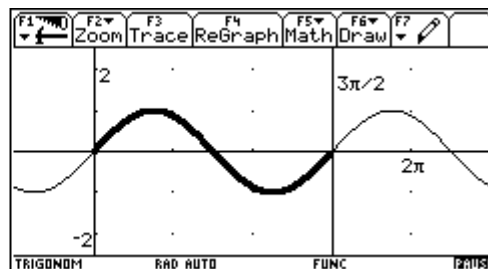
2

Text 27

Bei dieser Funktion beträgt die Länge der Periode $3\pi/2$.

Enter=OK ESC=CANCEL

-2



TRIGONDM RAD AUTO FUNC

F1 F2 F3 F4 F5 F6 F7

Zoom Trace ReGraph Math Draw

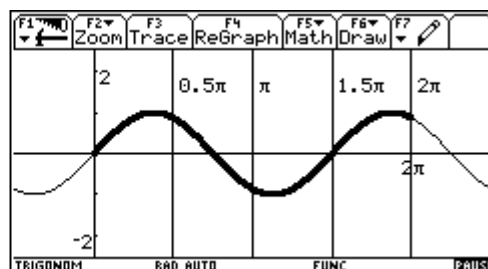
2

Text 28

Daher passen vier Drittel einer vollen Periode in das Intervall $[0, 2\pi]$.

Enter=OK ESC=CANCEL

-2



TRIGONDM RAD AUTO FUNC

F1 F2 F3 F4 F5 F6 F7

Zoom Trace ReGraph Math Draw

2

Zusammenfassung 3

Die Kreisfrequenz ω der Funktionen $f(x) = A \sin(\omega x + \phi)$ ist der Anteil der Periode, der auf das Intervall $[0, 2\pi]$ fällt

Enter=OK ESC=CANCEL

-2

TRIGONDM RAD AUTO FUNC

F1 F2 F3 F4 F5 F6 F7

Zoom Trace ReGraph Math Draw

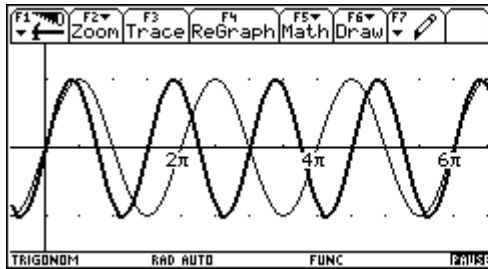
2

Text 31

Um den Bruchteil der Periode zu bestimmen, der innerhalb des Intervalles $[0, 2\pi]$ liegt, ist manchmal folgendes Vorgehen empfehlenswert. Betrachten wir die Funktion des letzten Beispiels noch einmal: $f(x) = \sin(4x/3)$ und vergleichen wir sie mit der Sinusfunktion.

Enter=OK ESC=CANCEL

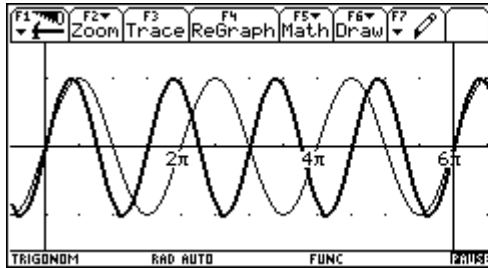
-2



Text 32

Man sucht die kleinste positive gemeinsame Nullstelle x_0 der beiden Funktionen, in der auch das Monotonieverhalten übereinstimmt. Diese Nullstelle ist in unserem Beispiel 6π .

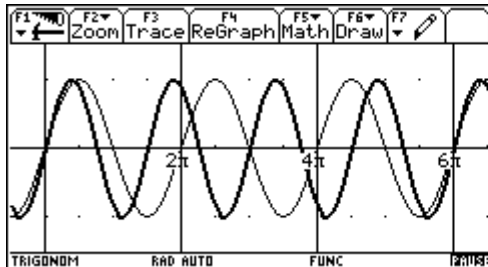
Enter=OK ESC=CANCEL



Text 33

Nun bestimmt man die die Anzahl a_1 der Perioden der Sinusfunktion im Intervall $[0, x_0]$, in unserem Fall 3.

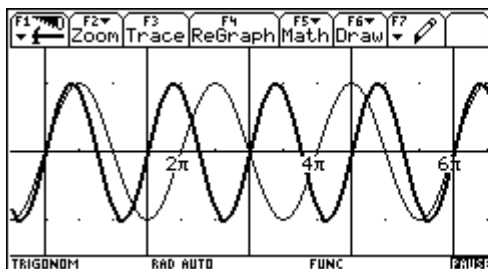
Enter=OK ESC=CANCEL



Text 34

Hernach bestimmt man die Anzahl a_2 der Perioden in Intervall $[0, x_0]$ der Funktion, deren Kreisfrequenz bestimmt werde soll, in unserem Fall 4.

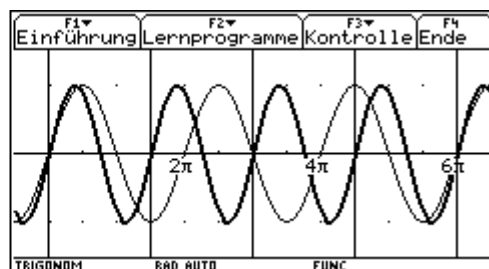
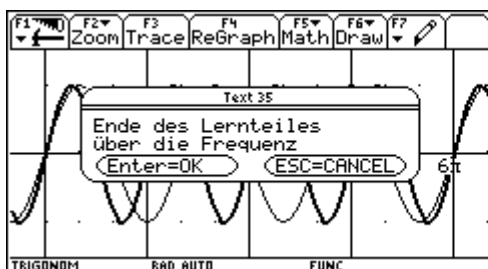
Enter=OK ESC=CANCEL



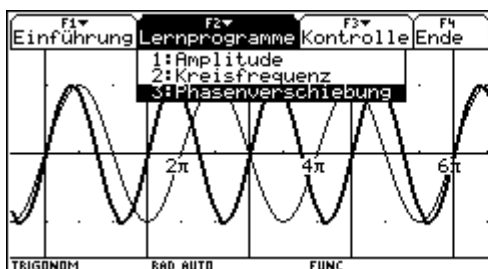
Text 35

Die Kreisfrequenz ω erhält man, indem man a_2 durch a_1 dividiert, also in unserem Fall $\omega=4/3$.

Enter=OK ESC=CANCEL



Zum Abschluß betrachten wir den Lernteil über die Phasenverschiebung.



Text 1

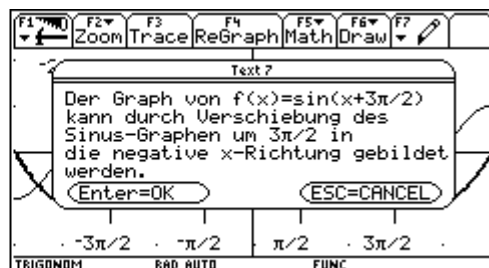
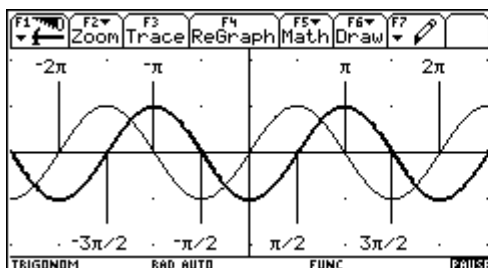
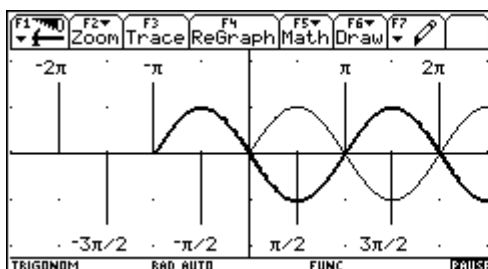
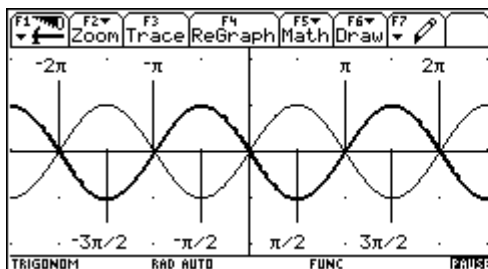
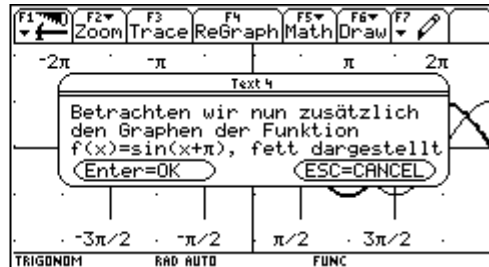
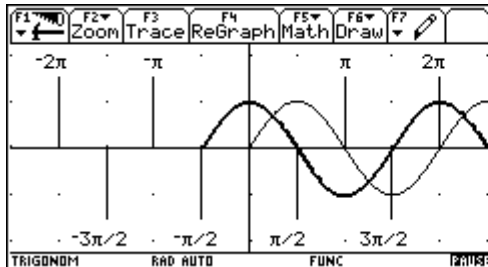
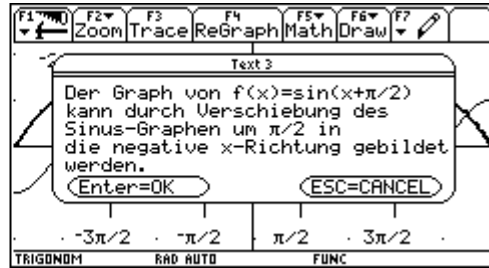
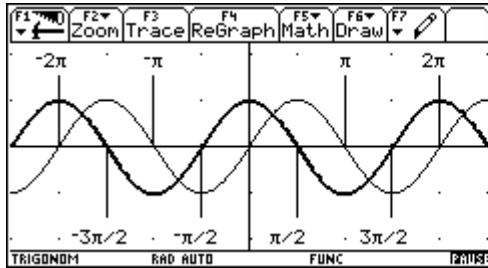
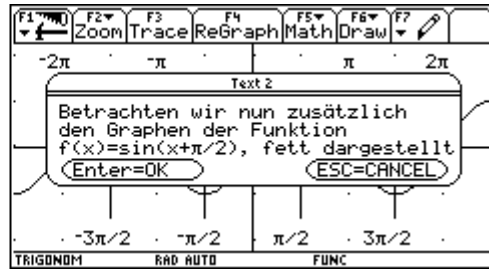
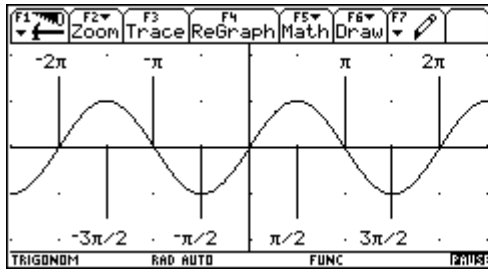
Beobachten wir zu Beginn den Graphen der Sinusfunktion.

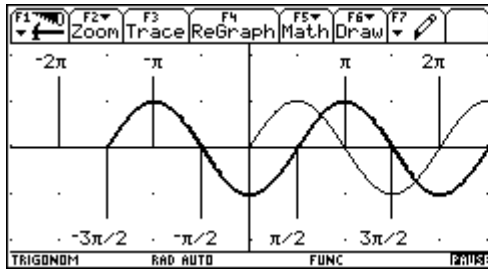
Enter=OK ESC=CANCEL

■ shell() Done

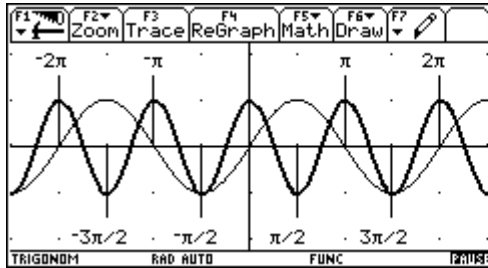
shell()

TRIGONDM RAD AUTO FUNC 1/30

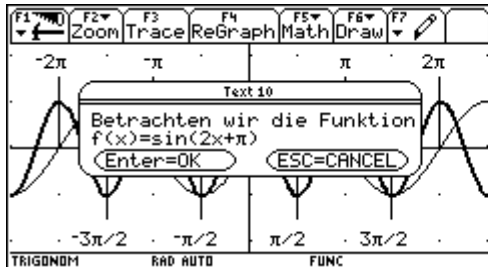




Text 8
 Wie wirkt sich die Phasenverschiebung nun in Zusammenhang mit einer Kreisfrequenz $\omega \neq 1$ aus? Betrachten wir die Funktion $f(x) = \sin(2x + \pi/2)$
 Enter=OK ESC=CANCEL



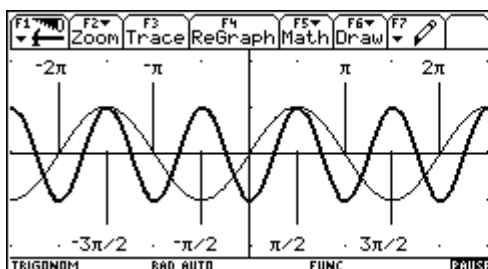
Text 9
 Periode und Verschiebung werden im gleichem Maß verkleinert. Der Graph besitzt bei $x=0$ sein Maximum.
 Enter=OK ESC=CANCEL



Text 10
 Betrachten wir die Funktion $f(x) = \sin(2x + \pi)$
 Enter=OK ESC=CANCEL



Text 11
 Periode und Verschiebung werden im gleichem Maß verkleinert. Der Graph besitzt bei $x=0$ eine Nullstelle, und ist bei $x=0$ monoton fallend.
 Enter=OK ESC=CANCEL



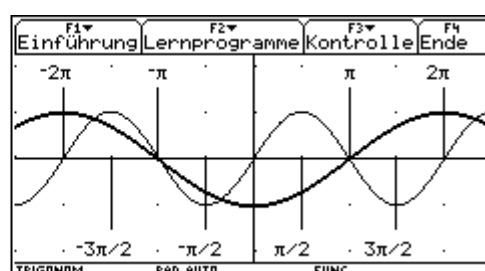
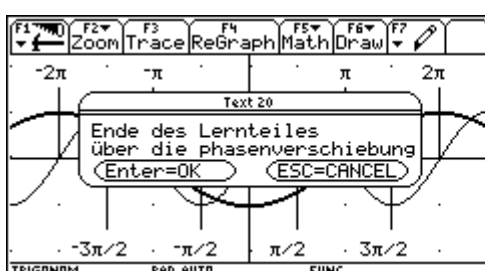
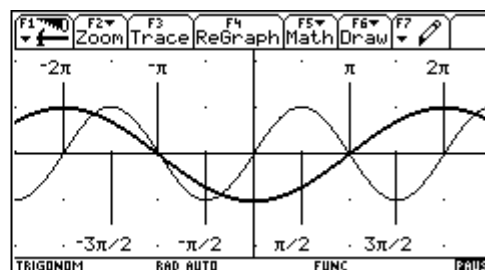
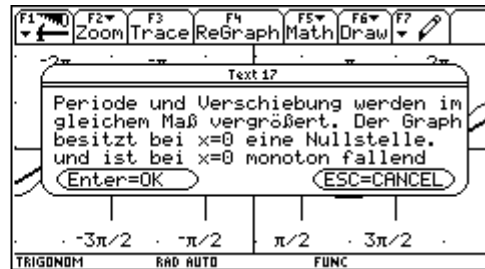
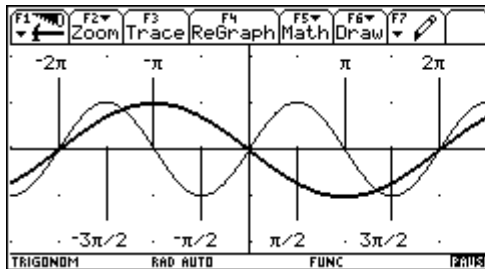
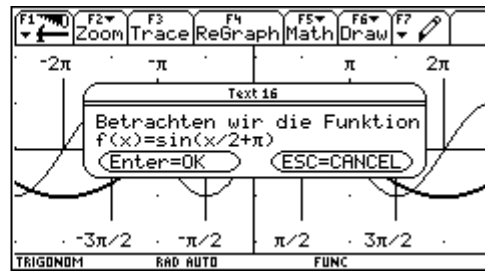
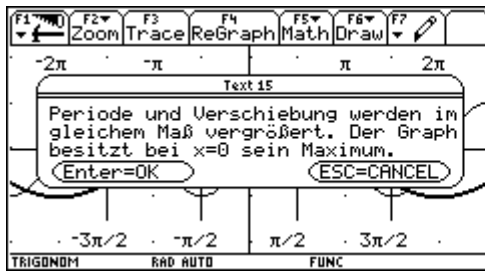
Text 12
 Betrachten wir die Funktion $f(x) = \sin(2x + 3\pi/2)$
 Enter=OK ESC=CANCEL



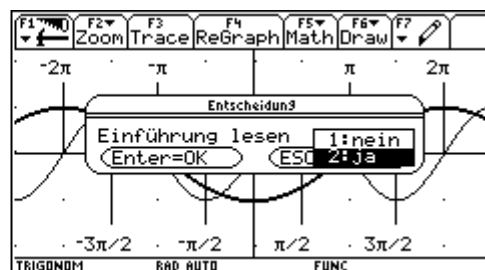
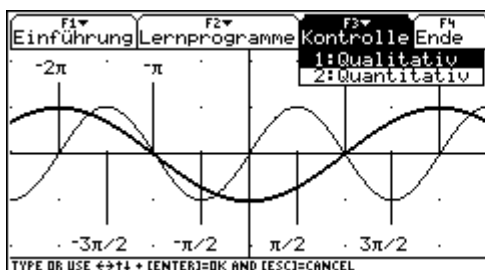
Text 13
 Periode und Verschiebung werden im gleichem Maß verkleinert. Der Graph besitzt bei $x=0$ ein Minimum.
 Enter=OK ESC=CANCEL

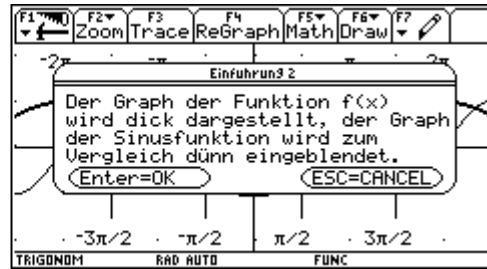
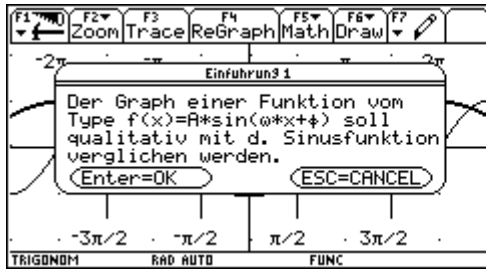


Text 14
 Betrachten wir die Funktion $f(x) = \sin(x/2 + \pi)$
 Enter=OK ESC=CANCEL

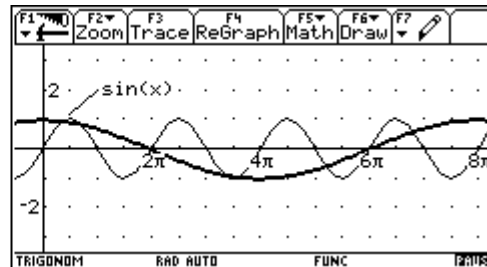


Nun wenden wir uns unter **[F3]** den Programmen zu, die unser Wissen überprüfen sollen. Dabei überprüfen wir zunächst qualitativ. Wieder gibt es eine Einführung zum Lesen.

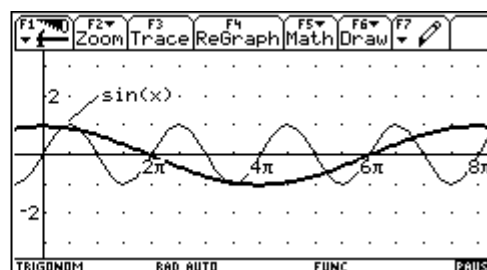
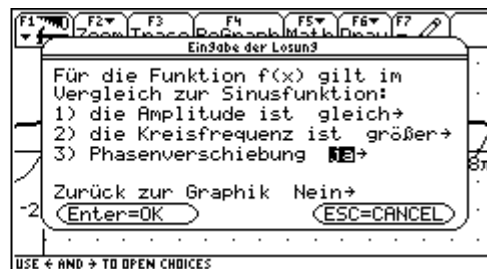
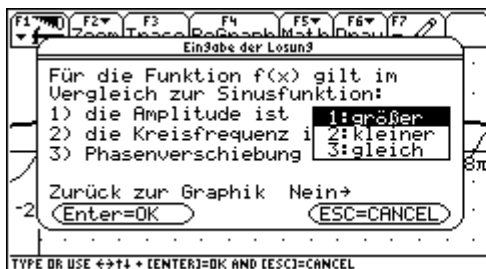




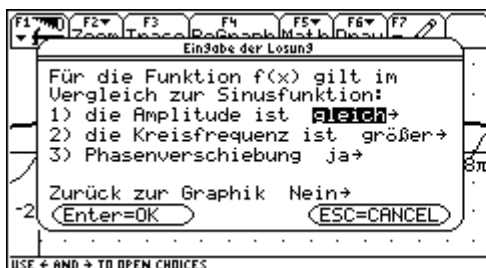
Danach startet sofort das Programm.

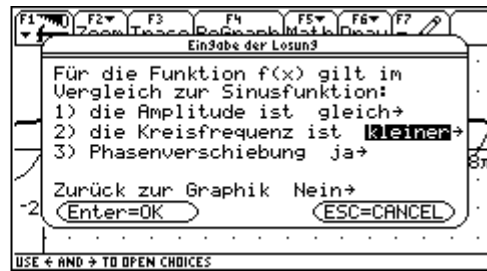
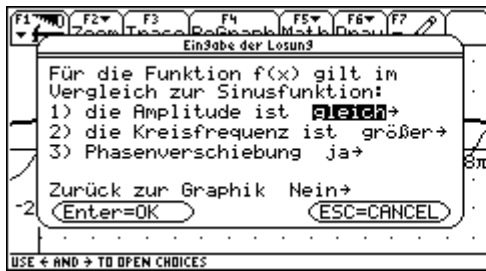


Wir treffen unsere Aussagen, wobei die Möglichkeit besteht zum Graphikbildschirm zurückzukehren.



Dann erhalten wir die Kontrolle durch das Programm. Es gibt nun die Möglichkeit das Programm zu beenden, die nächste Übung zu bearbeiten oder die Fehler auszubessern. Wir wollen die Fehler ausbessern.

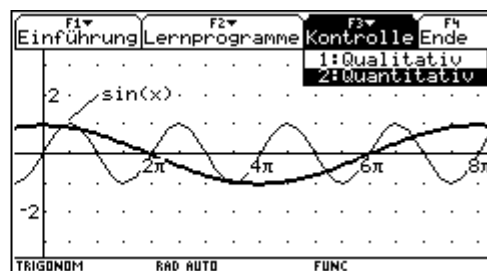
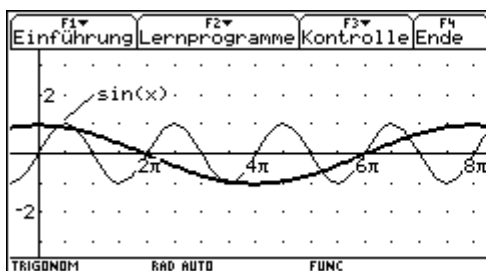




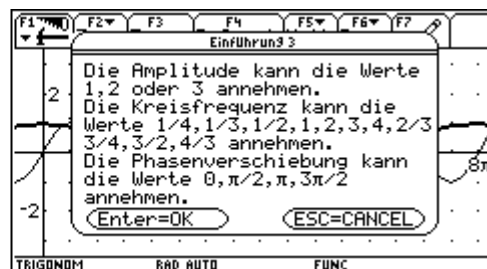
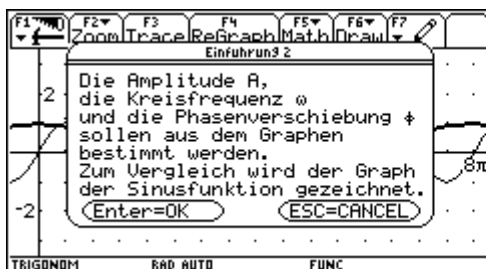
Mit richtiger Antwort wenden wir uns dem Programmende zu.



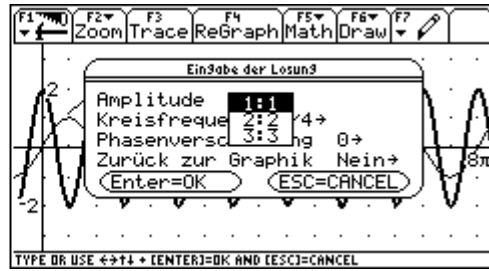
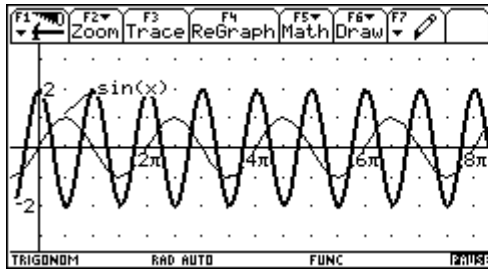
Dann starten wir die quantitative Kontrolle unseres Wissens.



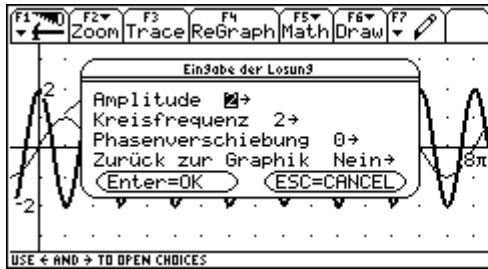
Wieder gibt es eine Einführung zum Lesen.



Danach startet sofort das Programm. Und wir machen unsere Eingaben.



Auch diesmal gibt es die Möglichkeit zur Graphik zurückzukehren. Wieder wird unsere Antwort vom Programm kontrolliert. Wir beenden das Programm.



Mit **[F4]** beenden wir die Programmgruppe.

