








Die Datei „FindeTerm“ ist im GeoGebra-Book <https://www.geogebra.org/m/vrszpydg> abrufbar und kann auf Smartphone, Tablet oder PC genutzt werden. Mithilfe der Anleitung kann die Erstellung nachvollzogen und variiert werden. Man erhält dabei einen ersten Einblick in interaktive Gestaltungselemente von GeoGebra. Fortgeschrittene Nutzer:innen können Varianten testen und eigene Erweiterungen implementieren.





Basisversion: Geraden zur Deckung bringen

Symbol	Inhalt / Beschreibung	... in der Eingabezeile:
Eingabezeile	Den rechts notierten Befehl in die Eingabezeile eingeben, um Steigung m_1 und y -Achsenabschnitt b_1 zu erzeugen.	$m_1=\text{Zufallszahl}(-4,4)$ $b_1=\text{Zufallszahl}(-4,4)$
Eingabezeile	Vorgegebene Funktion f definieren Mit dem ersten Befehl wird f definiert, mit dem zweiten Befehl wird der Graph von f schwarz gefärbt. Mit der Funktionstaste „F9“ kann man die Konstruktion aktualisieren, so dass eine neue Gerade generiert wird.	$f(x)=m_1*x+b_1$ SetzeFarbe(f ,schwarz)
	Schalter für „neue Aufgabe“ Man wählt das Werkzeug Schaltfläche aus und klickt an die gewünschte Stelle. Im sich öffnenden Menu trägt man den Namen „Neuer Graph“ und als Skriptzeile den Befehl AktualisiereKonstruktion() ein.	Schaltfläche("Neuer Graph") <i>Nach einem Rechtsklick auf die erscheinende Schaltfläche wählt man „Eigenschaften“ aus und trägt unter Skripting/ bei Mausklick den links zu sehenden Befehl ein.</i>
Eingabezeile	Zweite Funktion g für die Antwort erstellen Man definiert die lineare Funktion g , z.B. durch Eingabe von $g(x)=m*x+b$ und bestätigt die Einrichtung der Schieberegler m und b . Der Graph von g wird blau gefärbt, damit er sich vom vorgebenen Graphen von f unterscheidet.	$g(x)=m*x+b$ SetzeFarbe(g ,blau)
	Eingabefeld für m Nach Auswahl des Werkzeugs Eingabefeld klickt man ins Grafikkfenster und trägt als Beschriftung in der sich öffnenden Eingabemaske „Steigung m “ ein. Als verbundenes Objekt wählt man im Dropdown-Menu die Variable m aus.	Alternatives Vorgehen: <i>Eingabefeld(m)</i> [erzeugt ein Eingabefeld, das gleich mit m verknüpft wird] Nach Rechtsklick auf das Feld trägt man unter Eigenschaften / Grundeinstellungen die Beschriftung „Steigung m “ ein.
	Eingabefeld für b Ein zweites Eingabefeld wird wie oben angelegt. Als Beschriftung trägt man „ y -Achsenabschnitt b “ ein und verknüpft das Feld anschließend mit dem Objekt b .	<i>Eingabefeld(b)</i> mit der Beschriftung „ y -Achsenabschnitt b “
	Text-Werkzeug auswählen, auf die gewünschte Stelle im Grafikkfenster klicken und einen Aufgabentext eingeben, z.B.: „Bringe die blaue mit der schwarzen Gerade zur Deckung.“	Text1=“Bringe die blaue mit der schwarzen Gerade zur Deckung.“
	Testen Sie die Basisversion der Übungsumgebung, bevor Sie auf den folgenden Seiten mögliche Erweiterungen angehen.	




Erweiterung 1: Eingabefelder formatieren, Funktionsgleichungen einblenden

Symbol	Inhalt / Beschreibung	Hinweise
	Länge und Farbe der Eingabefelder anpassen Nach Rechtsklick auf ein Eingabefeld kann man es blau färben (im Register Farbe) und seine Länge von 20 auf 4 oder 5 Einheiten begrenzen (im Register Darstellung).	SetzeFarbe(Eingabefeld1,blau) <i>Die Länge wird danach im Eigenschaftsmenu wie links beschrieben angepasst.</i>
	Allgemeine Funktionsgleichung für f: $y=m \cdot x+b$ Nach Auswahl des Text-Werkzeuges klickt man an die gewünschte Stelle und gibt in der sich öffnenden Maske die Gleichung ein. Das Malzeichnen findet man unter „Symbole“.	Text2="y=m*x+b" <i>Das korrekte Malzeichen wird später im Register Text ergänzt.</i>
Eingabezeile	Konkrete Funktionsgleichung für g – Formeltext nutzen! Mit den rechts notierten Befehlen generiert man ein Textobjekt, das sich aus dem einleitenden Teil „y=“ und dem Formeltext der Funktion g zusammensetzt und färbt es blau. Der Befehl Formeltext ist ein äußerst nützlicher Befehl, mit dem sich das Erscheinungsbild der Funktionsgleichung steuern lässt.	Text3="y= " + Formeltext(g, true,false) SetzeWert(Text3,blau) <i>Hinweis: In neueren Versionen kann das oben vorangestellte "y= " ggf. wegfallen.</i>

Erweiterung 2: Rückmeldung mit bedingten Textobjekten und dynamischen Farben

Es gibt verschiedene Möglichkeiten, Rückmeldungen über richtige oder falsche Antworten einzublenden. Man kann z.B. das Textobjekt „Richtig“ genau dann anzeigen lassen, wenn die Antwort korrekt ist. In diesem Fall trägt man die Bedingung für die Sichtbarkeit des Textes im zugehörigen Eigenschaftsmenu unter „Erweitert“ ein. Hier wird ein weiterer Weg vorgestellt, bei dem die Texte dynamisch definiert werden.

Symbol	Inhalt / Beschreibung	alternative Eingabe
Eingabezeile	Dynamische Rückmeldetexte Man definiert den Text mit einer Wenn-Dann-Bedingung so, dass er in Abhängigkeit der Schülerantwort verschiedene Rückmeldungen zeigt. Die Textobjekte werden beim Ursprung eingeblendet, man platziert und formatiert sie nach Bedarf. <i>Anmerkung: Die Schreibweise „==“ mag verwundern, ist aber wichtig, mit nur einem Gleichheitszeichen funktioniert die Wenn-Dann-Bedingung nicht (vgl. Anleitung zu FindeGraph)</i>	Text4=Wenn(m==m_1,"Steigung stimmt","Steigung falsch") Text5=Wenn(b==b_1,"y-Achsenabschnitt stimmt", "y-Achsenabschnitt falsch") Text6=Wenn(f==g,"RICHTIG!") <i>Hinweis: Text 6 erscheint nur, wenn f=g gilt, sonst nicht.</i>
	Wechselnde Farben Das Textobjekt Text6 soll immer grün sein. Das Textobjekt Text4 soll bei richtiger Steigung grün, bei falscher Steigung rot erscheinen. In seinem Eigenschaftsmenu trägt man dazu im Register <i>Erweitert</i> die Bedingung für den Rot-, Grün- und Blauanteil ein (dabei an die Schreibweise "... == ..." denken). Entsprechend definiert man die Farbanteile von Text 5 mit der Bedingung $b==b_1$. Alternativ kann man auch in der Eingabezeile die Wenn-Dann-Bedingungen mit den nebenstehenden Befehlen komplett „übergeben“.	SetzeFarbe(<Objekt>,<Rot>,<Grün>,<Blau>) Farben setzen mit obigem Befehl SetzeFarbe(Text6,0,0.7,0) <i>Hinweis: Der Grünanteil 0.7 liefert einen etwas dezenteren Grünton als der Wert 1.</i> <u>Dynamische Farben definieren:</u> SetzeDynamischeFarben(Text4, Wenn(m==m_1,0,1),Wenn(m==m_1,0.7,0),0) SetzeDynamischeFarben(Text5, Wenn(b==b_1,0,1),Wenn(b==b_1,0.7,0),0)

Dynamische Farben

Rot:

Wenn(m \neq m_1, 0, 1)

Grün:



Wenn(m \neq m_1, 0.7, 0)

Blau:

0



Erweiterung 3: Zählvariablen für Antworten & Wechselschaltfläche zum Ein-/Ausblenden von g

Symbol	Inhalt / Beschreibung	alternative Eingabe
	<p>Wechselschaltfläche zum Ein-/Ausblenden Mit der neuen Schaltfläche 2 kann die Gerade g zur Differenzierung ein- oder ausgeblendet werden. Zunächst wird dafür das Kontrollkästchen w angelegt (und gleich verborgen, da es nicht sichtbar sein muss). Der Wahrheitswert w regelt die Sichtbarkeit der Gerade g.</p> <p>Alles kann mit den drei Befehlen rechts vom Skript der Schaltfläche aus gesteuert werden. Zunächst wird w negiert (Schreibweise !w = $\neg w$). Mit dem zweiten Skriptbefehl wird die Beschriftung der Schaltfläche aktualisiert, so dass sie wieder den nächsten möglichen Wechsel ausweist. Alternativ zur dritten Zeile könnte man die Bedingung w auch direkt im Eigenschaftsmenu von g (unter <i>Erweitert</i>) eintragen. Bei jedem Klick auf die Schaltfläche geht das Wechselspiel von vorne los.</p>	<p>Um hier mit einer Schaltfläche auszukommen, wird ein interessanter Trick im Skript der Schaltfläche 2 eingesetzt:</p> <pre>SetzeWert(w,!w) SetzeBeschriftung(Schaltfläche2, Wenn(w,"Gerade g ausblenden", "Gerade g einblenden")) SetzeSichtbarInGrafikansicht(g,1,w)</pre>
<p>Eine zusätzliche Motivation kann von der Rückmeldung zur Anzahl der in Folge korrekten Antworten ausgehen. Dazu benötigt man eine Zählvariable, deren Wert in einem Textfeld ausgegeben werden kann.</p>		
Eingabezeile	<p>Zählvariable "zaehler" definieren z.B. durch die Eingabe rechts als Schieberegler von 0 bis 50 mit Schrittweite 1. Blenden Sie diese Zählvariable in der Grafikanzeige aus, sie wird nur "im Hintergrund" benötigt.</p> <p>Aufeinander folgende korrekte Antworten zählen Mit den rechts notierten Befehlen wird im Skript der beiden Eingabefelder die Zählvariable "zaehler" jeweils um 1 erhöht, wenn die Antwort zum ersten Mal gegeben wird <i>und</i> richtig ist. Für die Steigung m in Feld E1 lautet die Zählbedingung z.B. $A_1 == 0 \ \&\& \ m == m_1$. In den beiden Antwortvariablen A_1 und A_2 wird gespeichert, ob eine Antwort schon gegeben wurde, indem der Variable dann der Wert 1 zugewiesen wird.</p> <p>Initialskript anpassen Damit alles funktioniert, muss noch das Skript der Schaltfläche "Neuer Graph" angepasst werden. Mit jeder neuer Aktualisierung werden die Zufallszahlen neu generiert und außerdem die Antwortvariablen (A_1, A_2) auf null gesetzt. Auch die beiden Eingabefelder werden zurückgesetzt, da m und b noch kein Wert zugewiesen wird. Die Funktion g wird dann in der Form $y = ?x + ?$ angezeigt.</p>	<pre>zaehler=schieberegler(0,50,1) Im Skript der Eingabefelder E1 (für m) bzw. E2 (für b) wird nun hochgezählt oder bei einer falschen Antwort auf null gesetzt: Skript von Eingabefeld E1 (für m): SetzeWert(zaehler,Wenn(A_1==0 && m==m_1,zaehler+1,0) SetzeWert(A_1,1) Skript von Eingabefeld E2 (für b): SetzeWert(zaehler,Wenn(A_2==0 && b==b_1,zaehler+1,0) SetzeWert(A_2,1) AktualisiereKonstruktion() SetzeWert(A_1,0) SetzeWert(A_2,0) SetzeWert(b,?) SetzeWert(m,?)</pre>
	<p>Rückmeldung im Textfeld - Ausgabe der Zählvariable Zur Motivation kann nun die Anzahl der aufeinander folgenden richtigen Antworten ausgegeben werden. Dazu bindet man das Objekt „zaehler“ in einem Textfeld ein (unter <i>Erweitert</i> / <i>GeoGebra-Symbol</i>).</p>	