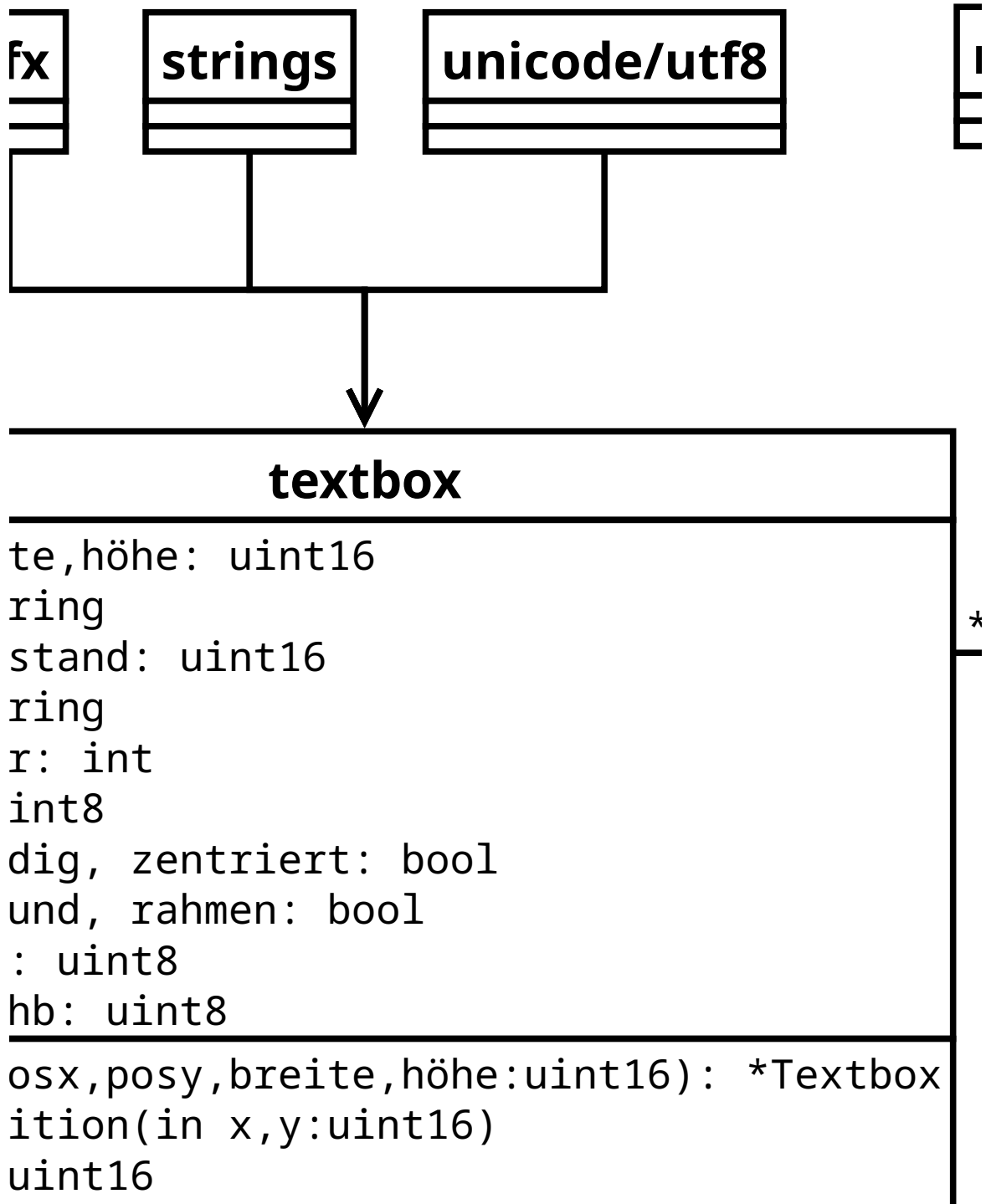
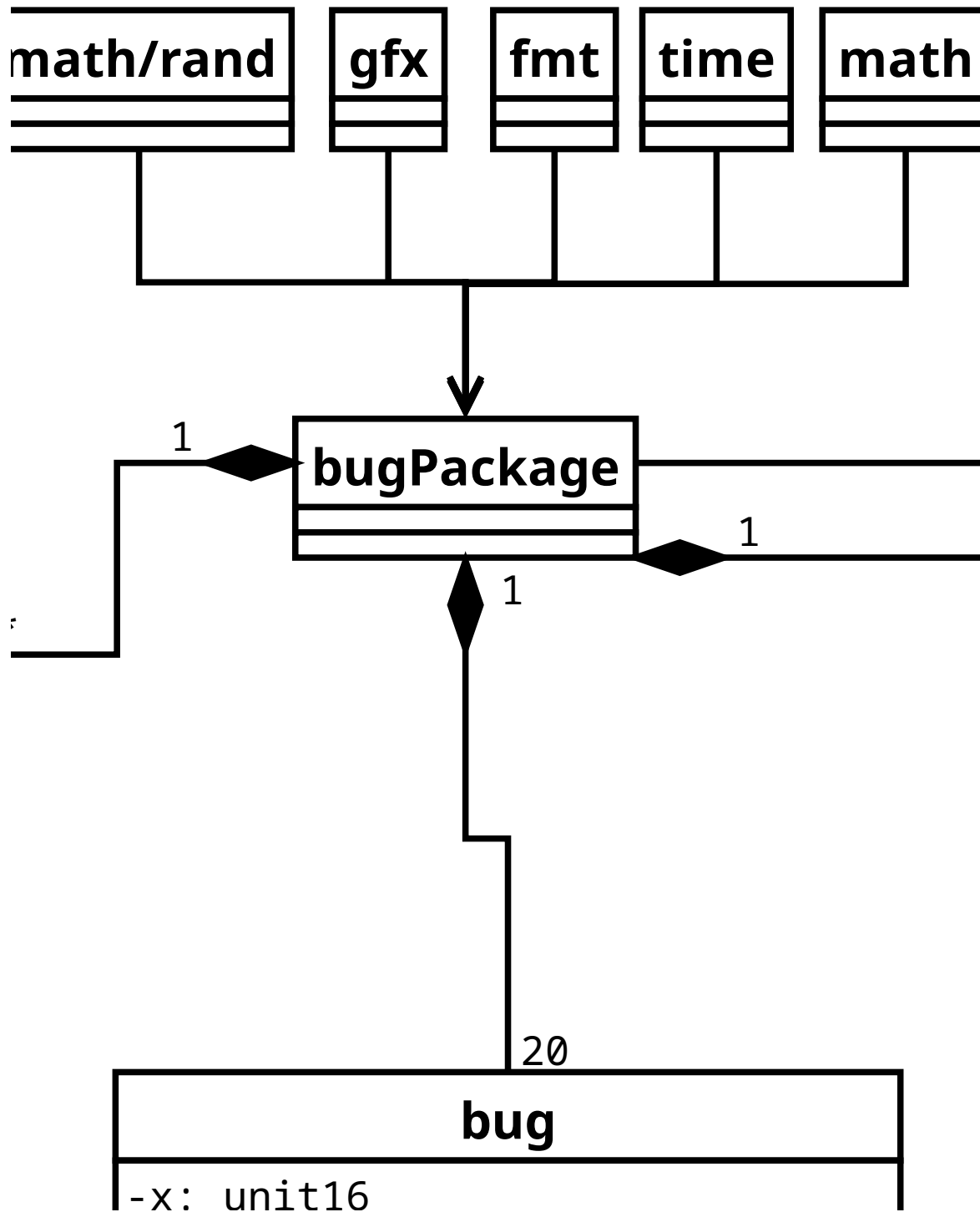


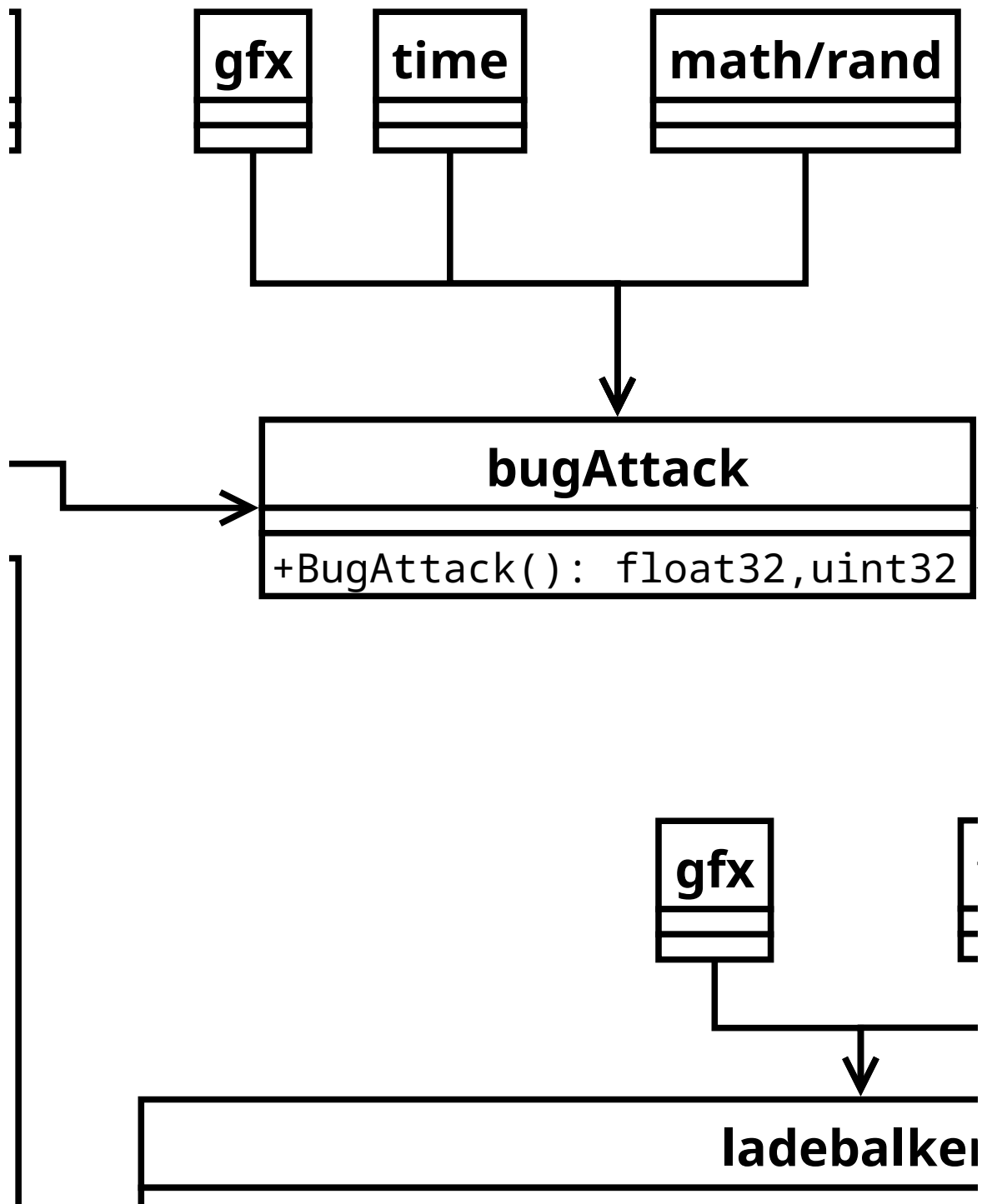
g1

```
-x,y,brei  
-font: st  
-zeilenAb  
-text: st  
-schriftg  
-r,g,b: u  
-linksbün  
-hintergr  
-rr,rg,rb  
-hr, hrg,
```

```
+New(in p  
+SetzePos  
+GibX():
```







```
+GibY():  
+SetzeBre  
+SetzeHöh  
+GibHöhe(  
+GibBreit  
+Schreibe  
+SetzeZei  
+GibZeile  
+SetzeFon  
+GibFont(  
+SetzeSch  
+GibSchri  
+SetzeFar  
+SetzeLin  
+SetzeZen  
+RahmenAn  
+SetzeRah  
+Hintergr  
+SetzeHin  
+Zeichne(  

```

```
uint16
ite(in b:uint16)
e(in h:uint16)
): uint16
e(): uint16
Text(in text:string)
lenAbstand(in za:uint16)
nAbstand(): uint16
t(in font:string)
): string
riftgröße(in gr:int)
ftgröße(): int
be(in r,g,b:uint8)
ksbündig()
triert()
(in r:bool)
menFarbe(in r,g,b:uint8)
undAn(in h:bool)
tergrundFarbe(r,g,b:uint8)
)
```

```
-y: uint16  
-alive: bool  
-dying: uint16  
-a: uint16  
-stirbt: bool  
-typ: uint8  
-speed: int  
-nervosität: int  
-r,g,b: uint8
```

```
+NewBug(in x,in y:uint16): *bug  
-zeichneBug()  
-eatCode()  
-startMoving()  
-bugAnimation()
```

		-wertAdresse : *uint16
		-x,y: uint16
		-r,g,b: uint8
		-taste: string
		-cdlänge: uint16
		-sound: string
		-newLadebalken(in wertAdresse:*uin y:uint16,in r,g,b:u in cd1:uint16,in sc
		-zeichne()
		-cooldown()