Průsečík přímky a kružnice s CAS Maxima

Zadání

Je dána kružnice $k: (x-m)^2+(y-1)^2=16$ a přímka $p: x-3y+6=0$. Určete hodnotu parametru *m*, aby přímka *p* byla tečnou ke kružnici *k*.

Teorie

Kružnice je zadána ve středovém tvaru, ze kterého pohodlně přečteme souřadnice středu [*m*;1] a poloměr přímky *r* = 4. Střed má tedy neznámou x-souřadnici (je dána parametrem *m*). Přímka je zadána obecnou rovnicí.

Abychom nalezli *m* (x-souřadnici středu), budeme hledat průsečíky *k* a *p* tak, aby existoval právě jeden (tzn. přímka s kružnicí mají jeden společný bod, je to tečna). Řešíme proto soustavu rovnic přímky a kružnice a hledáme společný bod [*x*;*y*].

Není nutné celou soustavu dořešit, stačí si uvědomit, že chceme, aby měla právě jedno řešení, proto ve chvíli, kdy dostáváme kvadratickou rovnici, stačí vzít její diskriminant a položit ho roven nule (je-li diskriminant nula, vyjde právě 1 řešení). Dořešíme jen rovnici s diskriminantem a dostáváme hodnoty parametru *m*.

Řešení

Spusťte program wxMaxima a začněte psát (i když nikde nebliká kurzor, to nevadí) nebo pište do okna online verze. Pokud používáte program wxMaxima, musíte za každým příkazem stisknout **Shift+Enter**, aby se vykonal!

<table>
<tr><th>Příkaz wxMaxima</th><th>Vysvětlení</th><th>Postup</th></tr>
<tr><td><code>k: (x-m)^2+(y-1)^2=16;</code></td><td>Vkládáme rovnici kružnice a pojmenováváme ji k.</td><td rowspan="2">Zadáváme rovnice do programu.</td></tr>
<tr><td><code>p: x-3*y+6=0;</code></td><td>Vkládáme rovnici přímky a pojmenováváme p.</td></tr>
<tr><td><code>solve([k,p],[x,y]);</code></td><td colspan="2">Necháváme program vyřešit soustavu rovnic k, p vzhledem k neznámým x a y, m je tedy parametr. Všimneme si, že x_1 a x_2, stejně jako y_1 a y_2 se liší přičtením či odečtením výrazu pod odmocninou (disktiminant), ten by tedy měl být 0. Zkusíme si nejprve vyřešit soustavu „ručně“ dosazovací metodou.</td></tr>
</table>

Příkaz wxMaxima	Vysvětlení	Postup
`solve(p,y);`	Necháme „vyřešit“ přímku *p* pro proměnnou *y* s „parametrem“ *x* neboli necháme vyjádřit z rovnice přímky *p* neznámou *y* vzhledem k *x*.	Vyřešíme soustavu rovnic *k* a *p* dosazovací metodou, dosazením za *y* z rovnice přímky.
`rhs(%[1]);`	Zobrazíme si jen pravou stranu rovnice, která nám vyšla, to dělá funkce `rhs()`. symboly `%[1]` říkají, že z předchozího výsledku, kterým byl (jednoprvkový) seznam (má hranaté závorky) bereme hned první (a zde jedinou) položku.	
`subst(%,y,k);`	Substituujeme (dosazujeme) do rovnice *k* místo neznámé *y* hodnotu předchozího výsledku (znak %). Pokud něco nefunguje, můžeme rovnou místo znaku % napsat `(x+6)/3`.	
`expand(%);`	Roznásobíme předchozí rovnici.	
`%-16`	Od předchozího výsledku odečteme 16, abychom dostali na levé straně kvadratický trojčlen a na pravé straně rovnice nulu.	Máme kvadratickou rovnici pro *x* vzhledem k *m*. Jelikož ale nepotřebujeme znát hodnotu *x*, ale pouze zajistit, aby mělo jediné řešení, budeme dále určovat diskriminant kvadratické rovnice tak, aby byl roven 0.
`lhs(%);`	Zobrazíme si jen levou stranu rovnice, tj. pouze kvadr. trojčlen.	
`ratcoef(%,x,2);`	Z předchozího výsledku (znak %) si zobrazíme koeficient pro *x* v 2. mocnině.	
`let(a,%o10);`	Nastavíme programu, že má písmeno *a* později substituovat hodnotou výsledku na řádku 10 (znaky %o10), tj. předchozí výsledek. Pozor, nelze použít znak %, výraz se bude vyhodnocovat až ve chvíli samotného substituování!	
`ratcoef(%o9,x,1);`	Z výsledku na řádku 9 (znak %o9, řádek s kvadratickým trojčlenem) si zobrazíme koeficient pro *x* v 1. mocnině.	
`let(b,%o12);`	Nastavíme programu, že má písmeno *b* později substituovat hodnotou výsledku na předchozím řádku (%o12).	
`ratcoef(%o9,x,0);`	Z výsledku na řádku 9 si zobrazíme koeficient pro absolutní člen (*x* v 0. mocnině)-	
`let(c,%o14);`	Nastavíme programu, že má písmeno *c* později substituovat hodnotou výsledku na předchozím řádku (%o14).	
`letsimp(b^2-4*a*c);`	Necháme vypsat vzorec pro diskriminant, funkce *letsimp* do něj dosadí za neznámé *a*, *b* a *c* dříve nastavené hodnoty.	
`%=0;`	V předchozím řádku určený diskriminant položíme roven nule.	Máme rovnici diskriminantu rovnu nule, hledáme pro jaká *m* bude nula a tudíž vyjde jediný průsečík přímky s kružnicí.

Příkaz wxMaxima	Vysvětlení	Postup
`solve(%);`	Necháme vyřešit rovnici v předchozím řádku.	Program nalezl 2 možné hodnoty parametru *m*. máme řešení.

Bude-li $m =\pm4 \sqrt{10} -3$, bude se kružnice dotýkat přímky.