

Profiling

V tomto dokumente sa nachádza tabuľka výsledkov profilovania aplikácie Kalkulačka a tiež zhrnutie získaných poznatkov z profilovania. V dokumente `vystup.pdf` sa nachádza ukážkový výstup profilingu z frameworku Android Studio. Boli vybraté ukážky predovšetkým nami implementovanými triedami, avšak v celkovom výpise profilingu sa nachádzali úplne všetky údaje (Vstavané funkcie, funkcie UI, ...). Z toho dôvodu sa výpis profilingu nachádza vo formáte PDF.

Class	Method	Invocation Count	Inclusive Time (µs)	Exclusive Time (µs)
MainActivity	ButtonOnClick	11	549 131	202
	sendToCalculate	3	70 504	118
	setEditAndTextView	8	478 379	790
	decBinHex	1	1 252	96
	deleteLastChar	9	290 074	828
	generator	2	96 888	146
	randInt	2	162	31
	isCharacterNumber	2	200	107
	Calculate	3	32 602	615
Compute	compute	3	4 806	408
	Validate	5	193	37
	exp	10	36	36
	make_better_guess	4	17	17
	sqrt	5	298	69
Parse	Parse	3	19 379	1 514
	isCharacterNumber	13	2 902	491
	isCharacterOperator	23	5 181	923
PostToInfix	convert	3	1 440	524
ProfilingActivity	readFromTxt	1	34 909	1 261
	showAlertDialog	1	270 422	596
	showDemo	1	309 012	1 023
	deleteEditText	1	47 478	23
StandardDeviations	comp	1	1 507	113
	Priemer	1	490	84

V tabuľke sú zelenou farbou znázornené metódy, v ktorých aplikácia trávi najviac času. Príčinou je vo väčšine prípadov práca s komponentami užívateľského rozhrania. Napríklad metódy ako `setEditAndTextView()`, `ButtonOnClick()`, či `deleteLastChar()` sme implementovali my, avšak narábame tam s funkciami vstavanými v operačnom systéme Android. Možnou optimalizáciou v týchto prípadoch by mohlo byť globálne definovanie widgetov UI s pridelením špecifického ID. Bolo by to možné v metóde `onCreate()`, ktorá sa nachádza v každej triede, ktorá má `extend AppCompatActivity`.

Metóda `readFromTxt()` načítava hodnoty pre smerodajnú odchylku zo súborov formátu TXT. V aplikácii sa nachádza aj možnosť zadania hodnôt pre smerodajnú odchylku priamo prostredníctvom UI. Metóda, ktorá to sprostredkúva sa nenachádza v tabuľke. Dôvodom je

veľmi malá dĺžka času tráviaceho v tejto metóde. Z toho nám vyplýva, že načítanie hodnôt zo súboru trvá podstatne dlhšiu dobu. Tento časový interval je avšak taktiež nepatrný a závisí aj od počtu hodnôt, ktoré sa v súbore nachádzajú. Pri desiatich hodnotách užívateľ ani nepozná väčšiu časovú odozvu. Pri hodnotách o počte tisíc a viac je už poznateľý rozdiel.

Na záver je potrebné spomenúť, že operačný systém Android je oproti napríklad iOS náročnejší na procesor a ak by bola aplikácia Kalkulačka implementovaná pre túto platformu, časová odozva aplikácie by bola nižšia.