

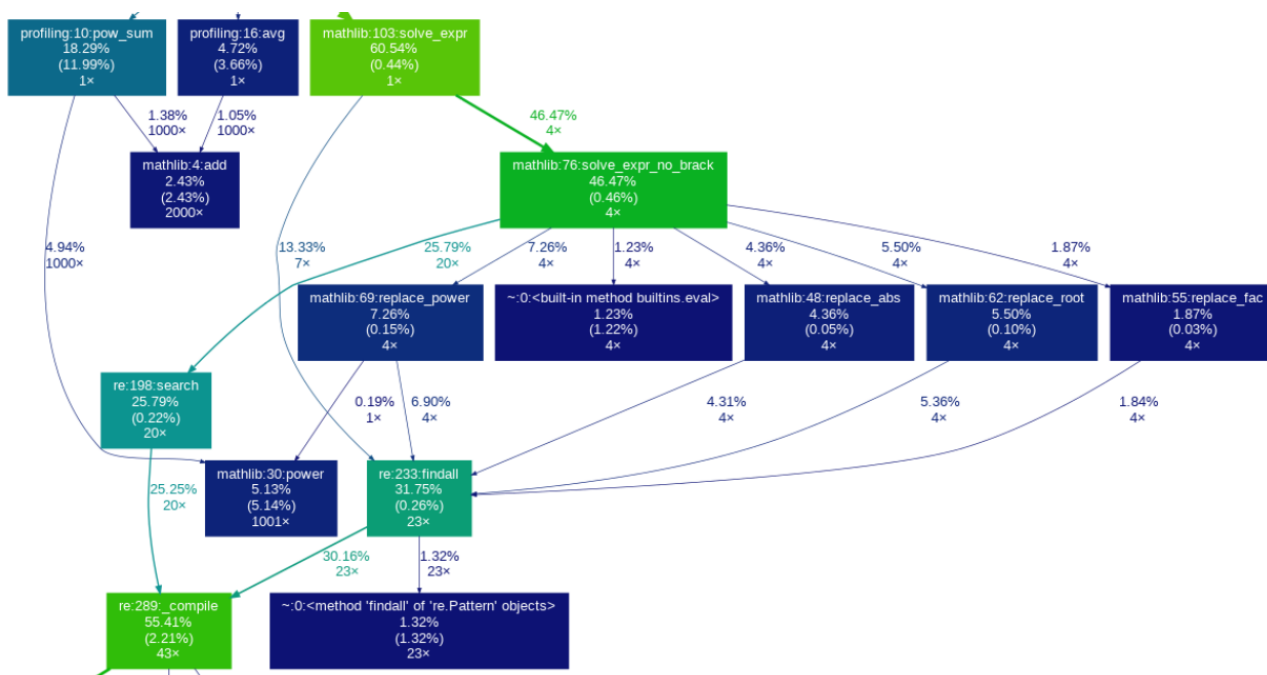
# Zpráva z profilingu

## IVS 2020/2021 projekt2 - kalkulačka

V profilingu sme sa zamerali na našu matematickú knižnicu **mathlib**. Naším cieľom bolo zistiť na akú časť knižnice sa zamerať pri optimalizácii.

Požili sme modul **cProfile**. Je to užitočný nástroj, ktorý zaznamená koľko krát bola volaná každá funkcia a koľko času bolo potrebného na jej vykonanie. Aby sme mohli tieto dáta premietnuť do grafu, použili sme vizualizér **gprof2dot** a **GraphViz**.

Za účelom profilingu sme vytvorili súbor **profilung.py**, ktorý počíta výberovú smerodajnú odchýlku. Vzorec pre túto odchýlku obsahuje veľa matematických funkcií a tak sme mohli efektívne otestovať všetky funkcie. Smerodajnú odchýlku sme postupne počítali z 10, 100, 1 000 a 10 000 čísel. Dostali sme tak 4 výsledky, ktoré sú priložené k tejto správe.



obrázok 1. výrez z profilingu 1000 čísel

Na obrázku môžeme vidieť výrez z jedného grafu. Čo je pre nás dôležité? Pozrieme sa na funkciu **solve\_expr**, ktorá zaberá viac ako 60% celkového času potrebného pre beh aplikácie. Avšak iba necelého 0.5% zaberá ona sama a zvyšný čas volá iné funkcie. Keď budeme postupovať po šípkach, zistíme, že veľa funkcií z našej knižnice používa knižnicu **re**. Z tejto knižnice je dôležité si pozrieť funkciu **re.findall** a **re.search**. Používanie práve týchto dvoch funkcií zaberá viac ako 55% času. Tu je jasné, že naša implementácia matematickej knižnice založená na hľadaní a nahradzovaní výrazov pomocou už vyššie spomenutej knižnice **re**, je jednoduchá a pri malom počte operácií efektívna. Nakoľko naša kalkulačka nikdy nebude spracovávať dlhé a na výpočet náročné vzorce, naše riešenie je dostačujúce. Ak by sme však chceli optimalizovať rozhodne by bolo na mieste porozmýšľať nad inou implementáciou. Rozumné by napríklad bolo využitie post-fixu a zásobníkovej aritmetiky.