

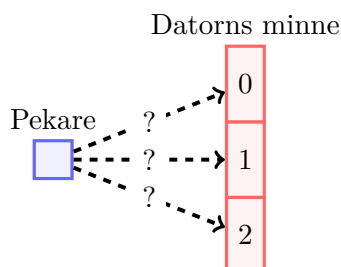
**EXAMENSARBETE** Pointer Analysis for Interactive Programming Environments**STUDENTER** Johan Arrhen, Ruben Wiklund**HANDLEDARE** Idriss Riouak (LTH), Anton Risberg Alaküla (LTH)**EXAMINATOR** Flavius Gruian (LTH)

# Snabb pekaranalys för textredigerare

POPULÄRVETENSKAPLIG SAMMANFATTNING **Johan Arrhen, Ruben Wiklund**

En pekaranalys kan användas för att ta reda på vilka platser i minnet en pekare kan peka till. I vårt examensarbete har vi programmerat en pekaranalys som går att köra snabbt nog för att den ska kunna användas medan en programmerare skriver kod.

Datorprogram innehåller variabler som kan läsas och skrivas till. En variabel kan innehålla till exempel ett tal eller ett ord. Datorer håller reda på vilket värde en variabel har genom att spara det i minnet. En speciell typ av variabler kallas för pekare. En pekare är en variabel som "pekar" till en plats i datorns minne. Pekare kan vara användbara när man behandlar data som behövs under lång tid.



## Vilka platser i minnet pekar pekaren till? Det kan en pekaranalys ta reda på!

Ibland kan det vara användbart att veta vilka platser i minnet en pekare kan peka till, utan att behöva köra programmet. Den informationen kan man ta reda på genom att utföra en pekaranalys på källkoden. Resultatet från en pekaranalys kan användas till att visa information som underlättar för en programmerare att skriva ett program. Till exempel så skulle datorn kunna varna om ett fel kan inträffa på en viss plats i programmet eller så

kan datorn ge förslag på vad programmeraren kan skriva på ett visst ställe. I sådana situationer är det viktigt att det går snabbt att köra analysen. Det skulle vara väldigt irriterande för programmeraren om hen var tvungen att vänta i en hel minut på att få ett förslag i sin textredigerare.

Många av de pekaranalyser som finns är antingen långsamma eller ger ett resultat där det står att en pekare kan peka till många platser som den i själva verket inte kan. I vårt examensarbete testade vi ett nytt sätt att göra pekaranalys på. Vår förhoppning var att det skulle både gå snabbt och ge ett bra resultat.

Vi prövade att göra en analys där man bara analyserar en viss del av koden istället för hela programmet. Hur stor del av programmet som analyseras kan justeras med en parameter  $k$ . Om man inte analyserar hela programmet riskerar man att missa viktig information i de delar som man inte analyserar, men genom att välja koden som analyseras på ett särskilt sätt så hoppades vi att det mesta av det som är relevant tas med. Så visades det sig också vara! Det gick att hitta 56% av resultatet på en femtedel av tiden som behövs för att köra analysen på hela programmet.

Eftersom analysen kan missa vissa saker så kan den inte användas till allt. Däremot passar den bra när man vill ha ett snabbt svar, vilket kan vara en fördel exempelvis i en textredigerare.