Софийски университет „Св. Климент Охридски“
Факултет по математика и информатика

ТЕМА ЗА ПРОЕКТ

към курс “Функционално програмиране”
за специалности Информатика, Компютърни науки (1 поток)
и Софтуерно инженерство
зимен семестър 2019/2020 г.

Извод на типове

Synopsis: Имплементирайте система за извеждане на типове в ламбда смятането.

Ламбда смятането (lambda calculus) е формална система в математическата логика, която описва изчисленията само на базата на абстракция (построяване) на функции и апликация (прилагане) на функции върху променливи и/или други функции чрез замяна на свързани променливи. Термовете (изразите) в ламбда смятането се построяват по следните правила:

- ако `x` е променлива, то `x` е терм. Приемаме, че разполагаме с изброимо безкраен списък с променливи.
- ако `M` и `N` са термове, то `(MN)` е терм, получен от прилагането на терма `M` над `N`. Прилагането е лявоасоциативна операция и когато пишем `MNP`, ще подразбираме `((MN)P)`.
- ако `M` е терм, а x е променлива, то `λx.M` е терм, получен като построяване на функция с аргумент x и тяло `M`. За удобство вместо `λx.λy.⋯` можем да пишем `λxy.⋯`.

Примери за ламбда термове: `λx.x`, `λx.y`, `λxy.x`, `λfx.f(fx)`, `λxyz.xz(yz)`

Едно от свойствата, които даден терм може да има, е тип. Типовете се построяват по следните правила:

- ако α е типова променлива, то α е тип
- ако σ и τ са типове, то σ→τ е тип на функция, приемаща аргумент от тип σ и връщаща резултат от тип τ. Операцията → е дясноасоциативна и когато пишем ρ→σ→τ, ще подразбираме (ρ→(σ→τ)).

Примери за типове: $\alpha \to \alpha$, $\alpha \to \beta$, $\alpha \to \beta \to \alpha$, $(\alpha \to \alpha) \to \alpha \to \alpha$

Важно свойство на типовете, които разглеждаме е, че те са *крайни*, правещи невъзможно съществуването на безкраен тип от рода на $\alpha \to \beta \to \gamma \to \cdots$. За целите на този проект можем да считаме, че няма и безкрайни ламбда термове.

Ще обозначаваме твърдението "термът M има тип τ" с $M{:}\,\tau$. За да определим типа на даден терм, първо е необходимо да направим някакви допускания за типовете на променливите, които участват в него. При различни допускания за променливите е възможно да се получи различен тип на терма. Затова ще разглеждаме твърдения от вида "при допускания Г термът M има тип τ", където Г е множество от допускания от вида $M{:}\,\tau$. Така можем да определим типа на произволен термов чрез следните правила

- ако $M{:}\,\sigma \to \tau$ и $N{:}\,\sigma$ при едни и същи допускания Г, то $(MN){:}\,\tau$ при същите допускания Г
- ако $M{:}\,\tau$ при допускания Г, сред които присъства и $x{:}\,\sigma$, то $(\lambda x.M){:}\,\sigma \to \tau$ при допускания Г с премахнато вече използваното допускане $x{:}\,\sigma$.

Примери: $xy{:}\tau$ при допускания $x{:}\,\sigma \to \tau$ и $y{:}\,\sigma$, $\lambda x.xy{:}(\sigma \to \tau) \to \tau$ при допускане $y{:}\sigma$, $\lambda yx.xy{:}\,\sigma \to (\sigma \to \tau) \to \tau$ без използването на допускания.

Целта на този проект е по даден ламбда терм, например $\lambda xy.x(xy)$ или $\lambda xyz.xz(yz)$, да се изведе неговия тип (ако има такъв) чрез типов извод. Извеждането на тип се получава по следната стратегия:

- за да намерим типа τ на терм от вида (MN), трябва първо да намерим такъв тип σ, такъв че $M{:}\,\sigma \to \tau$ и $N{:}\,\sigma$
- за да намерим типа на терм от вида $(\lambda x.M)$, можем да допуснем, че $x{:}\,\sigma$ за някой неизползван до момент тип σ и след това да потърсим типа на тялото M с добавеното ново допускане, че $x{:}\,\sigma$. Ако в процеса на търсене определим, че $M{:}\,\tau$, то ще знаем, че $(\lambda x.M){:}\,\sigma \to \tau$.
- в дъното на рекурсивното търсене ще ни се наложи да потърсим типа на променлива x; този тип можем да определим като погледнем в натрупаните допускания за предположение от вида $x{:}\,\sigma$.

Пример: За да намерим типа на терма $\lambda yx.xy$:

- допускаме, че $y{:}\,\sigma$
- за да намерим типа на терма $\lambda x.xy$:
  - допускаме, че $x{:}\,\rho$
  - за да намерим типа на терма xy:
    - търсим типа на терма y
      - но по допускане имаме $y{:}\,\sigma$
    - търсим типа на терма x от вида $\sigma \to \tau$

            - но по допускане имаме $x: \rho$
            - полагаме $\rho := \sigma \to \tau$
            - така получаваме $x: \sigma \to \tau$
        - така получаваме $xy: \tau$
    - така получаваме $\lambda x.xy: \rho \to \tau$
    - но $\rho = \sigma \to \tau$
    - затова получаваме $\lambda x.xy: (\sigma \to \tau) \to \tau$
- така получаваме, че $\lambda yx.xy: \sigma \to (\sigma \to \tau) \to \tau$

Примерно описание на идеята за типов извод и основните дефиниции в ламбда смятане можете да намерите [тук].