

Monády (a jejich kategorický duál – komonády) jsou důležitým konceptem teorie kategorií a zatímco monády jsou velmi oblíbenými nástroji ve funkcionálních jazycích (hlavně díky jazyku Haskell), komonády se příliš nepoužívají. V této práci prezentujeme definici komonád vhodnou pro potřeby funkcionálního programování a dále uvádíme příklady jejich praktického použití. Jedním z důležitějších příkladů je zipper – struktura, která reprezentuje určitou pozici. Ukážeme, že zipper lze automaticky odvodit pro každý regulární typ a také že tato operace připomíná derivaci z matematické analýzy. Kromě toho také uvádíme několik řešených příkladů v jazyce Haskell, které ilustrují, jak lze komonády použít pro řešení různých problémů. Všechny důkazy v teoretické části jsou provedeny v jazyce Agda a jsou zkontrolovány počítačem.