Začínající programátoři a praktické problémy

- nezkušenost při řešení běžných situací (jednání se zákazníkem, komunikace v týmu, hledání a oprava chyb,…)
- neznalost procedur při vývoji sw (jak se dělá analýza, návrh, dokumentace, jak testovat, …)
- podcenění procedur a metodik (neekonomické psaní kódu – hledání nejlepších „vyšperkovaných“ řešení, nedostatečná dokumentace, odbyté testování)

Metodika zaškolení začínajících pracovníků

- odpovídající metodiky prakticky neexistují
- je obtížné definovat, co by měli noví zaměstnanci umět
- metoda „odkoukání“ od zkušenějšího kolegy
    - zdroj problémů: zkušenější se cítí otravováni, méně zkušení podceňují => dlouhá doba „nepoužitelnosti“ začínajícího pracovníka
    - extrém – zkušenější jsou přetížení (a pak i často protivní), nemají žádný čas na nové kolegy, ti se naopak nemohou hnout z místa, neprodukují nic, jsou také otrávení a protivní

- setrvačnost tohoto přístupu – mnozí bývalí nováčkové se stejně chovají k nastupujícím pracovníkům
- rada zkušenějšího zabere ve výsledku podstatně méně času než pokusy nováčka se k řešení dobrat metodou pokus-omyl či pročítáním rozsáhlých dokumentací

„Nováčkem“ nemusíte být nutně pouze ve svém prvním zaměstnání, nováčkem se můžete stát i v pokročilém věku, pokud řešíte nové problémy v novém prostředí

Specifika IT

- v technologiích neustálé „nováčkovství“
- výhoda univerzálních znalostí (principy, způsob myšlení, ekonomie, psychologie), nebezpečí úzké (jinam nepřenositelné) specializace nepodložené univerzálními znalostmi