

Programski jezik F#

Seminarski rad u okviru kursa
Metodologija stručnog i naučnog rada
Matematički fakultet

Tijana Todorov, Tamara Garibovic,
David Nedeljkovic, Mihajlo Vicentijevic
tijana.todorov710@gmail.com, t.garibovic1995@gmail.com,
dneljkovic710@gmail.com, mihajlovicent@gmail.com

24. mart 2019

Sažetak

Dodati na kraju sažetak.

Sadržaj

1	Uvod	2
2	U 2018. godini F# opisan je u dokumentaciji kao "funkcionalni programski jezik koji se pokrece na .NET platformi" [5], ali odakle je on potekao?	2
2.1	Zasto je nastao jezik F#?	2
2.2	Koji programski jezici su najvise uticali na razvoj jezika F#?	3
3	Primena i mogucnosti	3
4	Zaključak	4
	Literatura	4
A	Dodatak	4

1 Uvod

Dodati na kraju uvod u temu i obavezno izmeniti trenutni radni naslov.

2 U 2018. godini F# opisan je u dokumentaciji kao "funkcionalni programski jezik koji se pokrece na .NET platformi" [5], ali odakle je on potekao?

Istorija programskog jezika F# datira jos od 1970. godine pa sve do danas. U ranim 70-im godinama na Univerzitetu u Edinburgu Robin Milner sa jos nekoliko svojih kolega razvija jezik ML (Meta-Language) baziran na programskom jeziku LISP. Njegova osnovna namena bila je pragmatičnog karaktera. Osmisljen je da pomogne u dokazivanju LCF (Logical computable functions) [4] teorema. ML jezik je koren svih jezika koji pripadaju familiji strogo tipiziranih funkcionalnih jezika, kao npr: Miranda, Haskell, Standard ML, Ocaml, EdinburghML, ReasonML, PureScript, a medju njima i F#.

Sve do danas ključne ideje ove familije jezika ostaju osnova jezika F# koji se nadograđivao kasnije iz dana u dan.

Tokom 80-ih godina dolazi do velike ekspanzije u kompjuterskoj industriji. Kako su se brzo razvijali softverski alati tako su se takodje pojavljivali novi jezici i programske paradigme. U tom periodu i Microsoft doživljava veliku ekspanziju kao kompanija koja razvija operativne sisteme i aplikacije. Medjutim, u ovom periodu, a najvise u kasnim 80-im godinama pojavljuje se novi, objektno-orijentisani talas razmišljanja u programiranju koji veoma utice na razvoj softvera.

Pocetkom 90-ih godina dok je Microsoft težio da održi monopol na tržištu i dok je u fokusu i dalje bila objektno-orijentisana paradigma, strogo tipizirani funkcionalni jezici bili su manje aktivni i okrenuti su ka razvitku na drugim poljima. Njihova primena uglavnom je imala ulogu u pronalazenju bagova, održavanju sistema i verifikaciji hardvera i softvera. Primeri jezika koji su se koristili za izgradnju ovakvih sistema su: Edinburgh ML, Standard ML, Ocaml, Caml Light, LISP. Takodje su neki funkcionalni jezici razvijeni u cilju istraživanja u okviru paralelnog programiranja, kao sto su Parallel ML i vrsta paralelnog Haskell-a [5].

2.1 Zasto je nastao jezik F#?

Tokom 90-ih godina Microsoft razvija .NET [3, 5] platformu za razvoj softvera. Ideja je bila da se omogući međusobna kompatibilnost više programskih jezika, odnosno da svaki jezik podržan na ovoj platformi može koristiti kôd nekog drugog jezika platforme. Glavni cilj Microsoft-a bio je da se na ovoj platformi podrži sto veći broj jezika iz različitih paradigmi.

U okviru ovoga Don Sym razvija projekat SML.NET koji je za ideju imao preusmeravanje Standard ML-a na .NET. Sistem je imao visok kvalitet, ali nije bio usvojen od strane programera. Pocetkom 2000-ih platforma .NET je već uveliko zazvela, ali za jezik SML.NET nije bilo većeg interesovanja. Autor je imao veliku želju da implementira strogo tipiziran funkcionalni jezik na način koji bi zainteresovao veliki broj programera. U

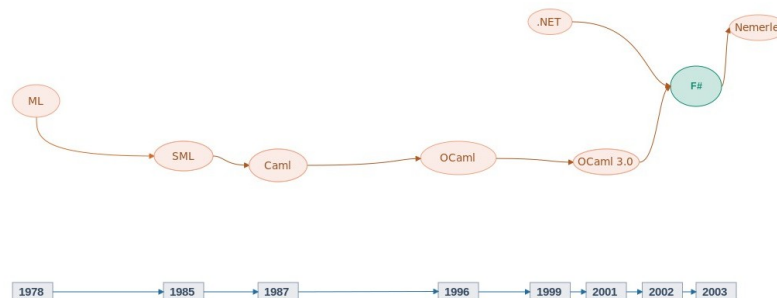
razmatranje ulazi i jezik OCaml, ali prethodno dolazi do pokusaja implementacije Haskell-a za .NET. Ovaj pokusaj bio je samo delimično uspešan i primenjen na malim programima. Rad na daljoj implementaciji je zauzastavljen.

U decembru 2001. godine Don Syme u razmatranje vraća jezik OCaml u cilju da se implementira za .NET platformu i razvija projekat Caml.NET koji će se kasnije preimenovati u F#.

Inicijalna ideja F# programskog jezika bila je jednostavna. Trebalo je da poveže prednosti OCaml programskog jezika sa prednostima .NET platforme. 2002. godine pojavljuje se prva verzija F# 0.5 koja bila je slabo primećena. Don Syme 2004. godine nastavlja intenzivno da razvija ovaj jezik, a početkom 2005. godine izbacuje prvu potpunu verziju ovog programskog jezika [5]. Poslednja aktuelna verzija ovog jezika je F# 4.1 [1].

2.2 Koji programski jezici su najviše uticali na razvoj jezika F#?

Najveći značaj za razvoj jezika F# imaju jezici SML (Standard ML) i CAML (Categorical Abstract Machine Language) porodica jezika koja je razvijana na Nacionalnom institutu za istraživanje informatike i automatizacije u Francuskoj 1994.godine [2]. U okviru ovog projekta pod nazivom “Cristal project” razvija se jezik Objective Caml koji ima veoma visoke performanse i naučnici ga koriste na Linux i MAC OS X platformama. Kao što je već ranije opisano, ovaj jezik i .NET platforma imali su najveći uticaj na razvoj jezika F#. Na slici 1 može se videti deo razvojnog stabla koji ovo prikazuje.



Slika 1: Razvojno stablo

3 Primena i mogućnosti

Snaga F# programskog jezika leži u svedenoj sintaksi koja omogućava laku čitljivost kôda, kao i efikasan razvoj programa koji zahtevaju primenu složenih matematičkih algoritama. Jezik omogućava brzo generisanje prototipova i njihovu brzu transformaciju u produkcionu kôd. Kôd napisan u F#-u lako se može paralelizovati, što je posebno značajno danas kada svi novi računari imaju više jezgara. F# danas ima široku primenu u obradi baza podataka, finansijskog modelovanja, statistici i bioinformatici. Takodje domen primene iz dana u dan raste.

F# je jezik koji ima pretežno funkcionalne karakteristike, ali on je svoju primenu pronasao u jos mnogo vrsta programiranja. Neki od njih su: imperativno, objektno-orjentisano, paralelno, distribuirano, asinhrono, meta programiranje, veb programiranje, skript programiranje, analiticko programiranje, i sl. Danas se on moze koristiti na velikom broju sistema, kao sto su: Linux, MAC, Windows, Android, iOS, i sl. O tome ce biti vise reci u nastavku.

4 Zaključak

Sta je zakljucak celog rada.

Literatura

- [1] F# Software Foundation and individual contributors. The F# Language specification, 2012-2018. on-line at: <https://fsharp.org/specs/language-spec/>.
- [2] Jon Harrop. *F# for Scientists*. Wiley-Interscience, New York, NY, USA, 2008.
- [3] Microsoft. .NET framework. on-line at: <https://dotnet.microsoft.com/learn/dotnet/what-is-dotnet>.
- [4] Robin Milner. Logic for computable functions: Description of a machine implementation. Technical report, Stanford, CA, USA, 1972.
- [5] Don Syme. The Early History of F#. on-line at: <https://fsharp.org/history/hopl-draft-1.pdf>.

A Dodatak

Ovde pišem dodatne stvari, ukoliko za time ima potrebe. Ovde pišem dodatne stvari, ukoliko za time ima potrebe. Ovde pišem dodatne stvari, ukoliko za time ima potrebe. Ovde pišem dodatne stvari, ukoliko za time ima potrebe. Ovde pišem dodatne stvari, ukoliko za time ima potrebe.