

Implementando soluções de Inteligência Artificial usando F#

Fabio Galuppo, M.Sc.

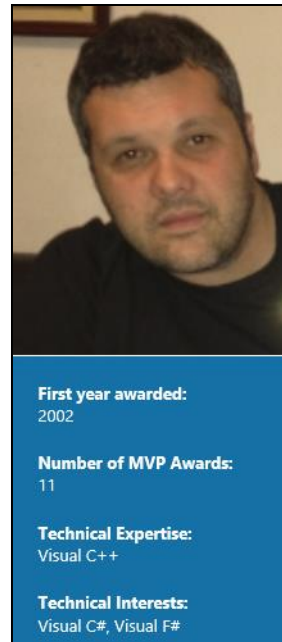
<http://fabiogaluppo.com>

fabiogaluppo@acm.org

Microsoft Most Valuable Professional (MVP)

Visual C++

2014



F#

At the core of every functional-first language is this:

simple, correct, robust code for solving complex problems

“Functional-first programming is a general-purpose programming technique particularly suited to tasks where Time-to-deployment, Efficiency, Correctness and Taming Complexity dominate.”

“Examples include ETL pipelines, general data-manipulation, calculation engines, service implementation, programmatic UIs and data science.

While these problems can be solved using other programming paradigms, they are particularly amenable to functional-first programming.”

Functional-first programming uses functional programming as the initial paradigm for most purposes, but employs other techniques such as objects and state as necessary.

Ref.: <http://www.slideshare.net/dsyme/fp-successv4redist>

Don Syme



archive.oredev.org

Inventions: F#

Don Syme is an Australian computer scientist and a Principal Researcher at Microsoft Research, Cambridge, U.K. He is the designer and architect of the F# programming language, described by a reporter as being regarded as “the most origi... +

en.wikipedia.org

On Windows, Microsoft recommend F# and the Visual F# tools for your functional-first programming needs

Programação Funcional

- » 1930s – Lambda calculus
 - Theoretical foundation of functional languages
 - Attempt to formalize all mathematics
- » 1958 – LISP
 - First functional (computer) programming language
- » 1978 – ML (meta-language)
 - Originally used in theorem proving systems
 - Useful as a general purpose language too!

- » 1990 – Haskell
 - Strict and lazy language, many advanced features
- » 1996 – OCaml (based on ML)
 - Combines functional and object-oriented features
- » 2002 – F# (based on OCaml)
 - Microsoft Research functional language for .NET
 - Now official part of Visual Studio 2010

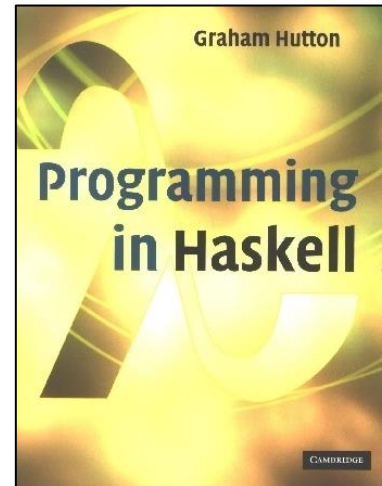
Ref.: <http://tomasp.net/materials/mff-fsharp-09/>

What is a Functional Language?

Opinions differ, and it is difficult to give a precise definition, but generally speaking:

- ⌘ Functional programming is style of programming in which the basic method of computation is the application of functions to arguments;
- ⌘ A functional language is one that supports and encourages the functional style.

Ref.: <http://www.cs.nott.ac.uk/~gmh/book.html>



Inteligência Artificial

- Major AI researchers and textbooks define the field as "***the study and design of intelligent agents***", where an intelligent agent is a system that perceives its environment and takes actions that maximize its chances of success. John McCarthy, who coined the term in 1955, defines it as "***the science and engineering of making intelligent machines***". Ref.: http://en.wikipedia.org/wiki/Artificial_intelligence

Basic Questions

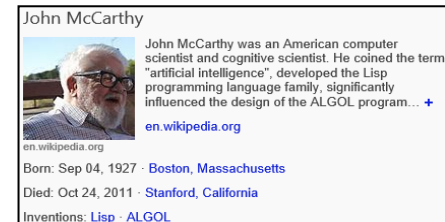
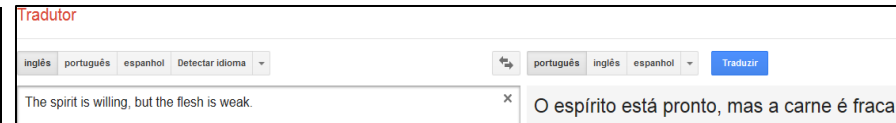
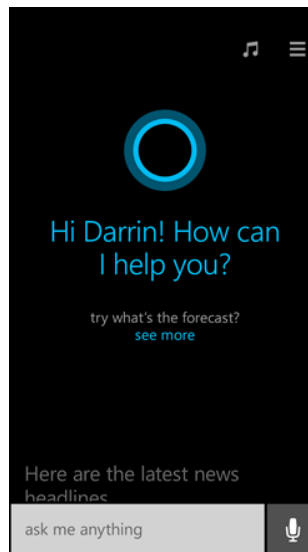
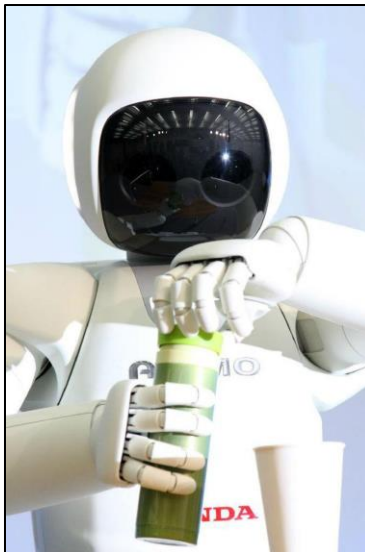
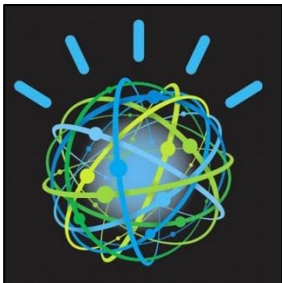
Q. What is artificial intelligence?

A. It is the science and engineering of making intelligent machines, especially intelligent computer programs. It is related to the similar task of using computers to understand human intelligence, but AI does not have to confine itself to methods that are biologically observable.

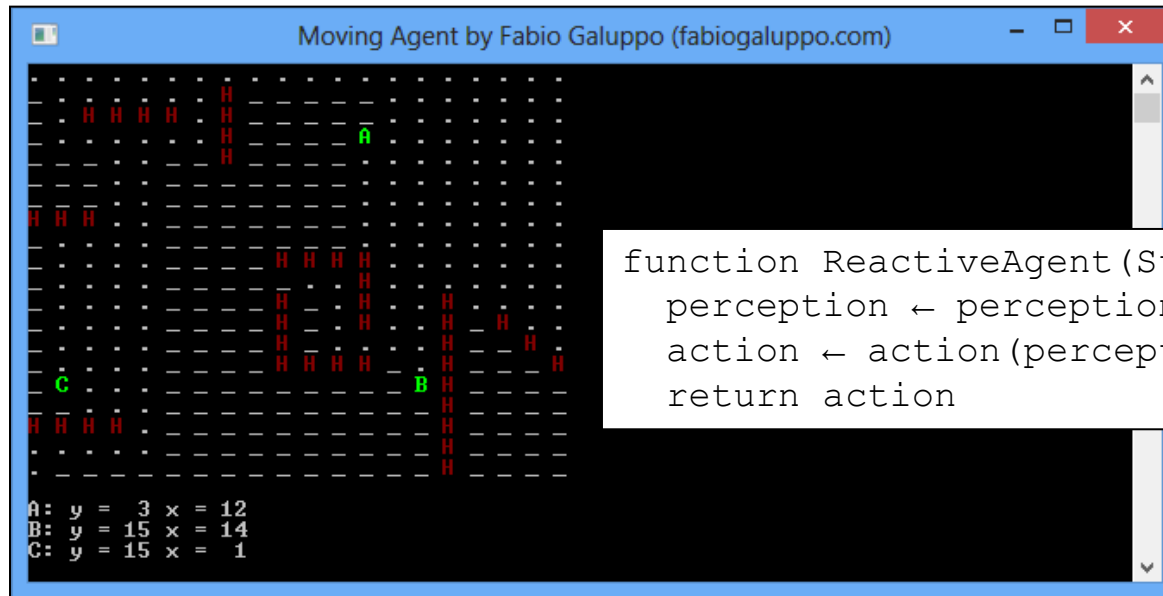
Q. Yes, but what is intelligence?

A. Intelligence is the computational part of the ability to achieve goals in the world. Varying kinds and degrees of intelligence occur in people, many animals and some machines.

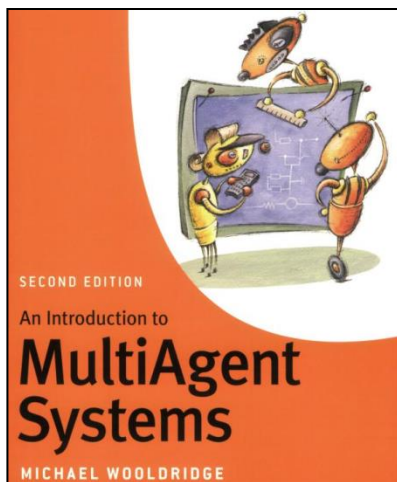
Ref.: <http://www-formal.stanford.edu/jmc/whatisai/node1.html>



Agentes relativos



```
function ReactiveAgent(State) : Action
  perception ← perception(state)
  action ← action(perception)
  return action
```



An Introduction to MultiAgent Systems
by Michael Wooldridge
ISBN-10: 0470519460
ISBN-13: 978-0470519462

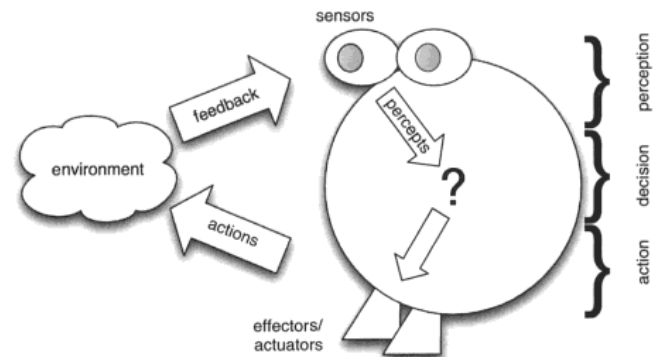


Figure 2.1: An agent in its environment (after [Russell and Norvig, 1995, p. 32]). The agent takes sensory input from the environment, and produces, as output, actions that affect it. The interaction is usually an ongoing, non-terminating one.

Agrupamentos de dados com *k-means*

Data Clustering

Detecting Abnormal Data Using k-Means Clustering

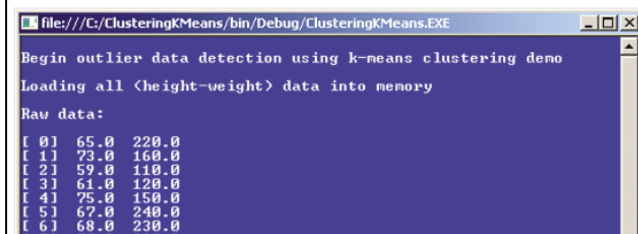
James McCaffrey

[Download the Code Sample \(VB Version\)](#)

Consider the problem of identifying abnormal data items in a very large data set, for example, identifying potentially fraudulent credit-card transactions, risky loan applications and so on. One approach to detecting abnormal data is to group the data items into similar clusters and then seek data items within each cluster that are different in some sense from other data items within the cluster.

There are many different clustering algorithms. One of the oldest and most widely used is the k-means algorithm. In this article I'll explain how the k-means algorithm works and present a complete C# demo program. There are many existing standalone data-clustering tools, so why would you want to create k-means clustering code from scratch? Existing clustering tools can be difficult or impossible to integrate into a software system, they might not be customizable to deal with unusual scenarios, and the tools might have copyright or other intellectual property issues. After reading this article you'll be able to experiment with k-means clustering and have the base knowledge to add clustering functionality to a .NET application.

The best way to get a feel for what k-means clustering is and to see where I'm headed in this article is to take a look at **Figure 1**. The demo program begins by creating a dummy set of 20 data items. In clustering terminology, data items are sometimes called tuples. Each tuple here represents a person and has two numeric attribute values, a height in inches and a weight in pounds. One of the limitations of the k-means algorithm is that it applies only in cases where the data tuples are completely numeric.



```
file:///C:/ClusteringKMeans/bin/Debug/ClusteringKMeans.EXE
Begin outlier data detection using k-means clustering demo
Loading all <height-weight> data into memory
Raw data:
[ 0] 65.0 220.0
[ 1] 73.0 160.0
[ 2] 59.0 110.0
[ 3] 61.0 120.0
[ 4] 75.0 150.0
[ 5] 67.0 240.0
[ 6] 68.0 230.0
```

MSDN Magazine – February 2013

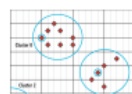
Ref.: <http://msdn.microsoft.com/en-us/magazine/jj891054.aspx>

James D. McCaffrey



For the American actor, see James McCaffrey. James D. McCaffrey is a software researcher and author known for his contributions to the fields of mathematical combinatorics and software test automation. McCaffrey holds a doctorate from...

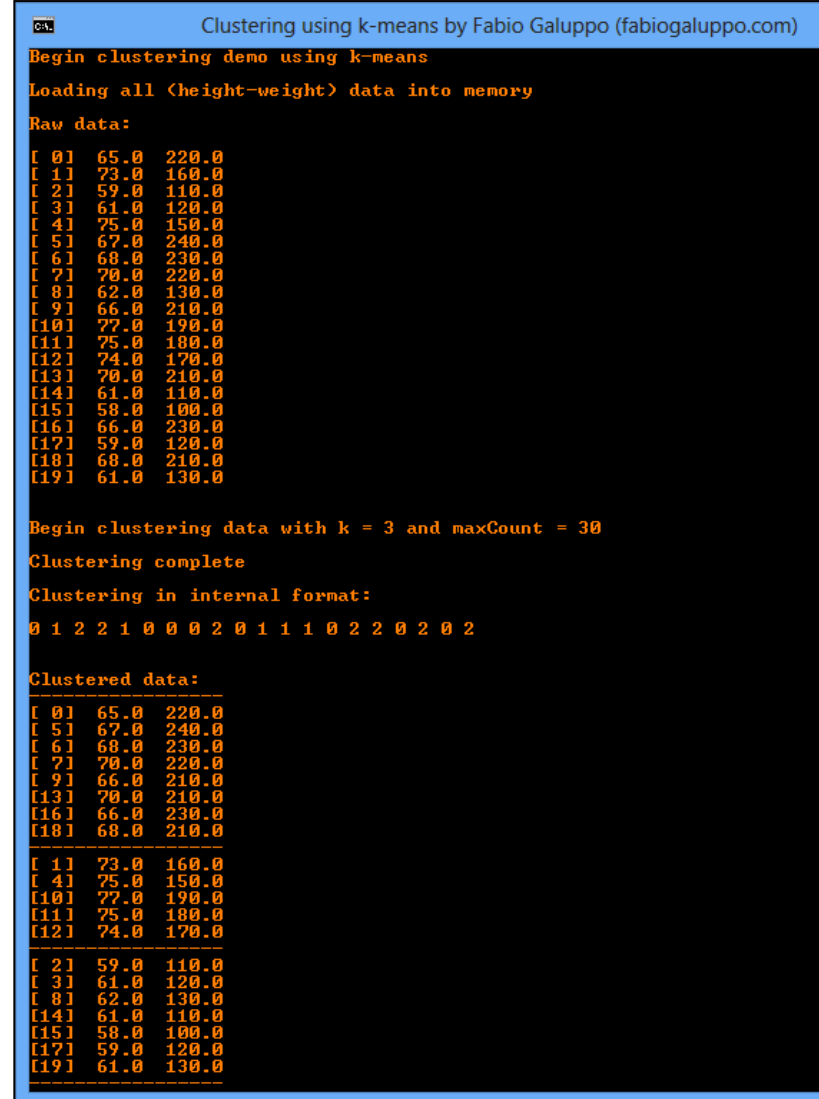
en.wikipedia.org



Data Clustering: Detecting Abnormal Data Using k-Means Clustering

Learn how C# developers can use k-Means clustering to group data items into similar clusters and enable detection of abnormal data.

James McCaffrey



```
Clustering using k-means by Fabio Galuppo (fabiogaluppo.com)
Begin clustering demo using k-means
Loading all <height-weight> data into memory
Raw data:
[ 0] 65.0 220.0
[ 1] 73.0 160.0
[ 2] 59.0 110.0
[ 3] 61.0 120.0
[ 4] 75.0 150.0
[ 5] 67.0 240.0
[ 6] 68.0 230.0
[ 7] 70.0 220.0
[ 8] 62.0 130.0
[ 9] 66.0 210.0
[10] 77.0 190.0
[11] 75.0 180.0
[12] 74.0 170.0
[13] 70.0 210.0
[14] 61.0 110.0
[15] 58.0 100.0
[16] 66.0 230.0
[17] 59.0 120.0
[18] 68.0 210.0
[19] 61.0 130.0

Begin clustering data with k = 3 and maxCount = 30
Clustering complete
Clustering in internal format:
0 1 2 2 1 0 0 0 2 0 1 1 1 0 2 2 0 2 0 2

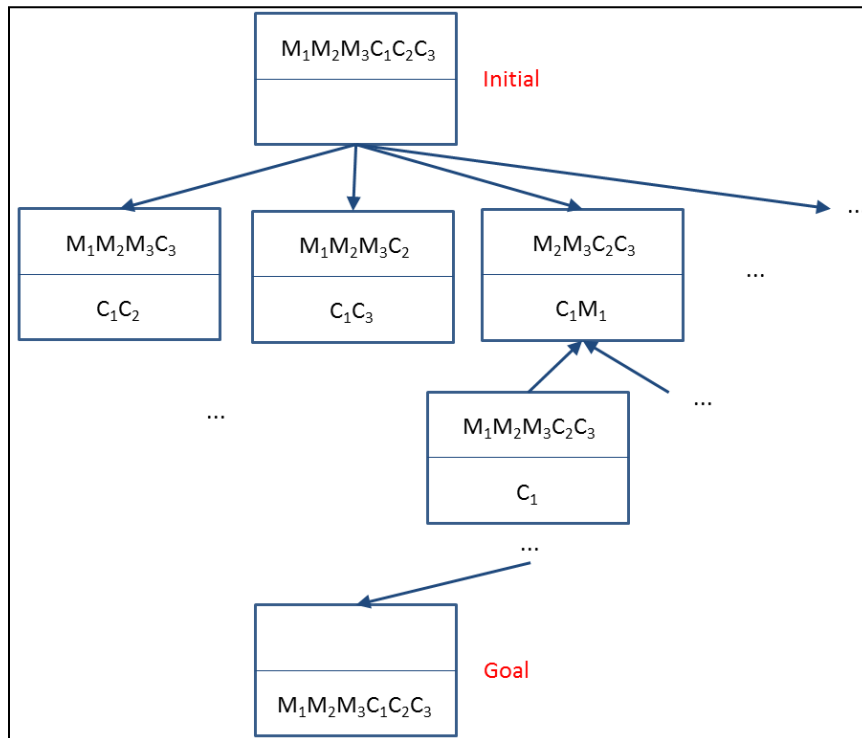
Clustered data:
[ 0] 65.0 220.0
[ 5] 67.0 240.0
[ 6] 68.0 230.0
[ 7] 70.0 220.0
[ 9] 66.0 210.0
[13] 70.0 210.0
[16] 66.0 230.0
[18] 68.0 210.0
[ 1] 73.0 160.0
[ 4] 75.0 150.0
[10] 77.0 190.0
[11] 75.0 180.0
[12] 74.0 170.0
[ 2] 59.0 110.0
[ 3] 61.0 120.0
[ 8] 62.0 130.0
[14] 61.0 110.0
[15] 58.0 100.0
[17] 59.0 120.0
[19] 61.0 130.0
```


Explorando o espaço de estados

- Problema dos canibais e missionários

- Puzzle clássico da Inteligência Artificial

- http://pt.wikipedia.org/wiki/Problema_dos_canibais_e_mission%C3%A1rios



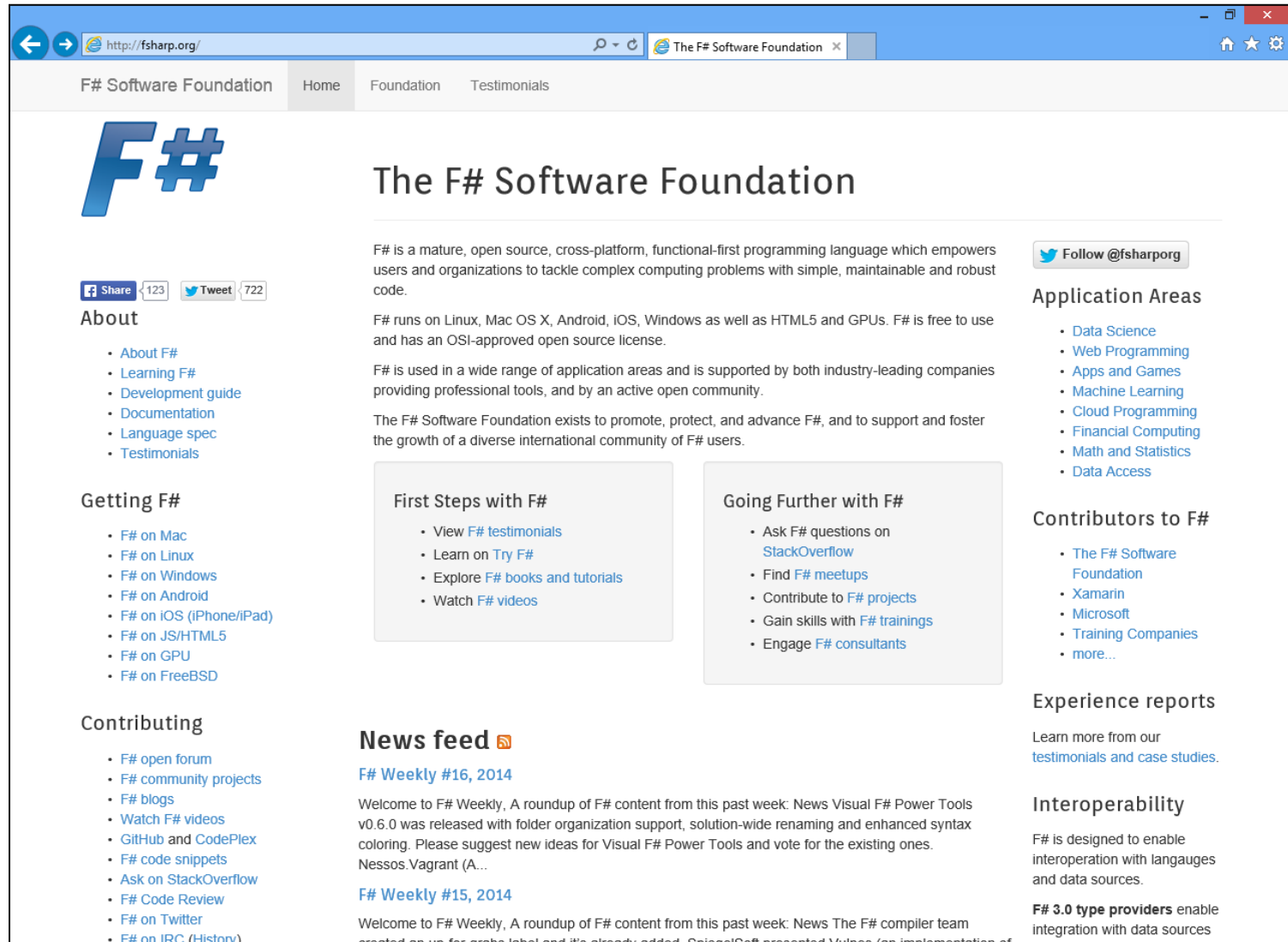
```

1: //state
2: type person = | M of int | C of int
3: type state = {top : Set<person>; bottom : Set<person>; isValid : bool}
4:
5: let initialState = {top = Set.ofList [C 1; C 2; C 3; M 1; M 2; M 3]; bottom = Set.ofList []; isV
6:
7: let goalState = {top = Set.ofList []; bottom = Set.ofList [C 1; C 2; C 3; M 1; M 2; M 3]; isVali
8:
9: //functions to problem solving
10: let isValid (xs:Set<person>) =
11:     let (c, m) = xs |> Set.fold (fun acc s ->
12:         let (c, m) = acc
13:         match s with
14:         | M _ -> (c, m + 1)
15:         | C _ -> (c + 1, m)) (0, 0)
16:     m = 0 || m >= c
17:
18: let moveDown (s:state) =
19:     let rec moveDownRec (t:Set<person>, b:Set<person>, xs:person list) =
20:         match xs with
21:         | x :: ys ->
22:             List.append [for y in ys ->
23:                 let bottom = Set.union b (Set.ofList [x; y])
24:                 let top = Set.difference t bottom
25:                 let isValid = isValid(top) && isValid(bottom)
26:                 {top = top; bottom = bottom; isValid = isValid}] (moveDownRec (t, b,
27:
28: moveDownRec (c tan c bottom (Set.toList c tan))
  
```

Ref.: <http://fssnip.net/mA>

F# Software Foundation

<http://fsharp.org/>



The screenshot shows the F# Software Foundation website in a web browser. The browser's address bar displays 'http://fsharp.org/'. The website has a navigation bar with 'Home', 'Foundation', and 'Testimonials'. The main content area features the F# logo, a 'Follow @fsharporg' button, and sections for 'About', 'Getting F#', 'Contributing', 'News feed', 'Application Areas', 'Contributors to F#', 'Experience reports', and 'Interoperability'. The 'About' section describes F# as a mature, open source, cross-platform, functional-first programming language. The 'Getting F#' section lists platforms like Mac, Linux, Windows, Android, iOS, JS/HTML5, GPU, and FreeBSD. The 'Contributing' section lists ways to get involved, such as through forums, community projects, blogs, videos, GitHub, CodePlex, code snippets, StackOverflow, Code Review, Twitter, and IRC. The 'News feed' section shows two entries for 'F# Weekly' from 2014. The 'Application Areas' section lists fields like Data Science, Web Programming, Apps and Games, Machine Learning, Cloud Programming, Financial Computing, Math and Statistics, and Data Access. The 'Contributors to F#' section lists organizations like The F# Software Foundation, Xamarin, Microsoft, and Training Companies. The 'Experience reports' section mentions testimonials and case studies. The 'Interoperability' section discusses integration with languages and data sources, specifically mentioning F# 3.0 type providers.

F# Software Foundation Home Foundation Testimonials

The F# Software Foundation

F# is a mature, open source, cross-platform, functional-first programming language which empowers users and organizations to tackle complex computing problems with simple, maintainable and robust code.

F# runs on Linux, Mac OS X, Android, iOS, Windows as well as HTML5 and GPUs. F# is free to use and has an OSI-approved open source license.

F# is used in a wide range of application areas and is supported by both industry-leading companies providing professional tools, and by an active open community.

The F# Software Foundation exists to promote, protect, and advance F#, and to support and foster the growth of a diverse international community of F# users.

Follow @fsharporg

About

- [About F#](#)
- [Learning F#](#)
- [Development guide](#)
- [Documentation](#)
- [Language spec](#)
- [Testimonials](#)

Getting F#

- [F# on Mac](#)
- [F# on Linux](#)
- [F# on Windows](#)
- [F# on Android](#)
- [F# on iOS \(iPhone/iPad\)](#)
- [F# on JS/HTML5](#)
- [F# on GPU](#)
- [F# on FreeBSD](#)

Contributing

- [F# open forum](#)
- [F# community projects](#)
- [F# blogs](#)
- [Watch F# videos](#)
- [GitHub and CodePlex](#)
- [F# code snippets](#)
- [Ask on StackOverflow](#)
- [F# Code Review](#)
- [F# on Twitter](#)
- [F# on IRC \(History\)](#)

First Steps with F#

- View [F# testimonials](#)
- Learn on [Try F#](#)
- Explore [F# books and tutorials](#)
- Watch [F# videos](#)

Going Further with F#

- Ask F# questions on [StackOverflow](#)
- Find [F# meetups](#)
- Contribute to [F# projects](#)
- Gain skills with [F# trainings](#)
- Engage [F# consultants](#)

Application Areas

- [Data Science](#)
- [Web Programming](#)
- [Apps and Games](#)
- [Machine Learning](#)
- [Cloud Programming](#)
- [Financial Computing](#)
- [Math and Statistics](#)
- [Data Access](#)

Contributors to F#

- [The F# Software Foundation](#)
- [Xamarin](#)
- [Microsoft](#)
- [Training Companies](#)
- [more...](#)

News feed

F# Weekly #16, 2014

Welcome to F# Weekly, A roundup of F# content from this past week: News Visual F# Power Tools v0.6.0 was released with folder organization support, solution-wide renaming and enhanced syntax coloring. Please suggest new ideas for Visual F# Power Tools and vote for the existing ones. Nessos.Vagrant (A...

F# Weekly #15, 2014

Welcome to F# Weekly, A roundup of F# content from this past week: News The F# compiler team created an up-for-grabs label and it's already added. SpiegelSoft presented Vulpes (an implementation of

Experience reports

Learn more from our [testimonials and case studies](#).

Interoperability

F# is designed to enable interoperation with languages and data sources.

F# 3.0 type providers enable integration with data sources

It's all about Polyglot Programming!



C++ supports systems programming. This implies that C++ code is able to effectively interoperate with software written in other languages on a system. The idea of writing all software in a single language is a fantasy. From the beginning, C++ was designed to interoperate simply and efficiently with C, assembler, and Fortran. By that, I meant that a C++, C, assembler, or Fortran function could call functions in the other languages without extra overhead or conversion of data structures passed among them.

<http://www.youtube.com/watch?v=NvWTnloQZj4>



Bjarne Stroustrup: The 5 Programming Languages You Need to Know

“Nobody should call themselves a professional if they only knew one language.”

...**C++**, of course; **Java**; maybe **Python** for mainline work... And if you know those, you can't help know sort of a little bit about **Ruby** and **JavaScript**, you can't help knowing **C** because that's what fills out the domain and of course **C#**. But again, these languages create a cluster so that if you knew either five of the ones that I said, you would actually know the others...

“Inclua a esta lista **F#**, **Scala**, **Haskell**, **Erlang**, **Clojure**, **Lua** e/ou **Racket**” – Fabio Galuppo

Implementando soluções de Inteligência Artificial usando F#

Fabio Galuppo, M.Sc.

<http://fabiogaluppo.com>

fabiogaluppo@acm.org

Microsoft Most Valuable Professional (MVP)

Visual C++

2014

