

The Programmer's Oracle

Explanation of The Deck

To start our history with this deck, Hillary and I both have experiences with coding. While she works in Computer Science for a living, I do more casual work, primarily leaning towards design. Her experience is a career, while mine is a hobby.

One groggy February morning, over Skype while we were both at work, Hillary paused and sent me a message claiming it was "the doofiest idea ever" about making a programmer's Tarot deck. The example she gave, Missing Semicolon, is in this deck.

I followed it up with the statement: "The Programmer's Oracle, because sometimes you just really need to see that shit coming," in reference to easily overlooked mistakes while coding. Then, a moment later, we both started getting ideas. That was the start of The Programmer's Oracle.

We finally settled on 28 cards for this oracle deck. At first, we were trying to brainstorm more cards without creating redundancies, but then, mid-call, we both stopped talking to think about it. Both 4 and 7, in numerology, call to the organized, the scientific, the logical. A moment later, we agreed. The deck didn't need more cards. It was perfect.

A couple of months later, I present to you the finished work, The Programmer's Oracle. A combination of divination and computation, a blend of magic and technology.

April 2016

Amanda Sieracki

Witchdraft.tumblr.com

Team Hivemind Productions

How this PDF works

Definitions will be the technical explanations for each term/card. **Meanings** are what would generally come with a divination deck. While they sometimes share explanations with the definitions, they are meant to be looser, for purposes of divination. Some cards have little easter eggs in the design. For the most part, our cards are designed with classic coding elements in mind, hence the black backgrounds, and green or white text. Of course, this doesn't carry over to all cards.

We also only provide a guide to reading the cards. Your interpretations will always be just as valid as ours. Divination is an art that is more up to the diviner than any guide. As you grow more familiar with the deck, feel free to include your own definitions, and feel free to use the blank card provided for a meaning you may have thought should be added, or to signify the unknown. We'd love to hear what you come up with!

The Cards

Functional Program

Definition: A functional program is a predetermined state that will always return that exact same state. If the function has been set to always be the number 4, anytime you call upon that function, it will provide the number four. Nothing else. By declaring the function, you know what it will always be.

Meaning: A self-fulfilling prophecy. You have determined that the future is set, and you know what it is set as. By saying this, your actions will subconsciously alter things around you to make it come true. People say that negativity breeds negativity. If you declare today is awful and will not change from being awful, you project that onto everything you do, and make it awful, thus proving your point. On the other hand, if you declare it's a great day, all your actions will further that positivity, and you will make your own day better.

GOTO

Definition: A GOTO command is as literal as it sounds. It tells the code that it needs to go to a place. However, it's so old and outdated, most programmers cringe at the knowledge that it exists in something they may need to work on or update. It can cause significant problems and errors in places, as it has to be read in order.

Meaning: It may seem like the most direct route, but it doesn't mean it's the best. In this case, it's a long, bad, and dangerous way to your goal. Rethinking your path may be needed.

Main()

Definition: The start, the beginning. The root that branches into other functions. It also is the absolute last part of a program. All commands and functions need to finish before this one can.

Meaning: A beginning. The start of something, of anything. It's not just a vague start, but the defining moment of one. Everything further in on your journey will be reminiscent of this moment.

For Loop

Definition: A repeated command which runs a specific number of times. They're versatile and almost always useful. They can count up or down.

Meaning: A limited repetition. As much as things seem to be a constant repeat, there will come a time where it will end, and move on.

While Loop

Definition: A loop that continues until a true statement becomes false. If the statement never becomes false, it never ends. Everything that a For Loop can do, a While Loop can do, but can also do more.

Meaning: Newton's First Law of Motion states that an object at rest stays at rest, and an object in motion stays in motion with the same speed, and in the same direction, unless acted upon by an unbalanced force. For this card, this means things will continue in the same vein until something changes it. If you are in

a positive cycle, you will stay that way until a negative external force acts on you. On the other hand, if you are in a negative cycle, only a positive force will change things.

Missing Semicolon

Definition: So many programming languages can give this error. This makes it so the compiler can't find the end of a statement and the code will compile wrong. Hard to catch error.

Meaning: A missing piece that would clarify a situation. It's often something small, easily overlooked, but can be that one thing that can really mess things up or make things unclear. Once found, however, what last looked like one path may very easily become two.

Buffer Overflow

Definition: The wrong data is accessed, and messes up everything. Can also display a sense of vulnerability. This happens because of a non-null terminated string or array, usually. It lets the process reading the data continue on into further registers without being stopped.

Meaning: A miscalculation that causes an unfamiliar or uncomfortable situation, reminiscent of The Tower in standard Tarot decks. It's not necessarily something that can be fixed, or if it can, not easily.

Segmentation Fault

Definition: Caused by trying to read or write an illegal memory location. It's usually the result of misallocated data, or trying to read past the end of an array.

Meaning: At first glance, this card seems terrible. Something is about to go wrong, or has already, and it looks bad. However, once things settle down, it can be fixed. It's only a temporary stop along the line, and worrying too hard about it may prevent you from seeing the solution.

Natural Language

Definition: Human language instead of computer language. It can also be when an interpreter or compiler can understand human language, like in text adventure games.

Meaning: A clear answer, simplicity in understanding. Sometimes communication, just straight-forward words, are all you need.

Machine Learning

Definition: Teaching the computer through target goals, repetition, and failure. It's like teaching a child to learn. You can set a goal, and variables, and run it through a program so it can self-edit the variables until it learns.

Meaning: Learning from experience, making mistakes to learn. Sometimes it's the only way to learn what does or doesn't work, and to progress from there.

Queue

Definition: A list with a definite order. There are many different kinds, but every queue has an order.

Meaning: Don't worry about how long it'll take. **Get** in line, and wait your turn. It'll arrive, right when you need it to.

Patch

Definition: A minor release, to fix a bug. They generally do not contain features, but rather focused on fixing bugs.

Meaning: A remedy, a fix. It may also be needing to fix a problem at hand, rather than attempting to add something new. Sometimes you need to work on the old stuff before adding a new thing.

Variables

Definition: A variable is a container that holds a value, a piece of data that can change as the program runs.

Meaning: An answer that changes as needed. What may have once been false may now be true, and vice versa.

Permissions

Definition: Both what a user is or isn't allowed to do, and what any program is or isn't allowed to do or access.

Meaning: Ability to enter into something. You have to be able to say "I can do this". Sometimes the only thing not giving you permission to, is you.

Globals

Definition: Variable accessible from anywhere within the program, by any object/function

Meaning: Access no matter where you are. There's nothing standing in your way, no matter where you stand.

Objects

Definition: It's anything, and all the methods that can be used to manipulate or get more information about the thing. Its self-contained information and the ways it can be manipulated.

Meaning: Self-knowledge and awareness, either in the general or the incredibly specific.

Unbound Loop

Definition: Loop where the boundary conditions are unable to be met or have never existed. This results in a situation in which the program will never end, and keep consuming memory until it consumes even the memory that the operating system needs to run.

Meaning: An unending loop that can consume everything around it. It is a good card to use to represent chronic mental or physical illness.

Symbolic Link

Definition: A good example of this is to look at your computer desktop. Those shortcuts are symbolic links, calling up the program when you click on them.

Meaning: A logical shortcut. It's like walking down an alley connecting two streets rather than walking around. You won't miss anything important, you're just skipping past unnecessary things.

Architectural Diagram

Definition: Used to plot out structures of a program with inputs and outputs, and any processes that need to be run on those inputs and outputs. A plan for your program to work.

Meaning: Setting up a plan in which to follow, writing your own fate. Whether this is a ten year plan, or just a general daily schedule, it's a guide for you to follow.

Test Cycle

Definition: No one likes to be the one who needs to test all the work, but you need it for things to go as you expect.

Meaning: Grunt work. The necessary evil to catch problems before they start. Everyone has something like this they have to do, but you'll regret it more by not doing it.

Bug

Definition: This is why you need test cycles. Bugs, named for an early computer mishap, are unexpected errors that can pop up in the most unlikely of places. Sometimes bugs can be useful, or fun. Sometimes they go on to become features.

Meaning: A hitch in the road, can be good or bad.

Public

Definition: Anything within a program can access the data in anything that is public.

Meaning: Open to others, being on display.

Private

Definition: Can only be accessed within its own scope (ie, its own class or own function)

Meaning: Hidden, personal. Keeping things quiet.

Compiler

Definition: A compiler is like a translator for programs, to translate from human readable to what the computer can use.

Meaning: Translation for clarification in communication. Sometimes words get muddled, or just may come across wrong. Sometimes there needs to be another point, words need to be simplified.

Constants

Definition: A Constant is like a variable, but the value is set at declaration and can't change ever, no matter what.

Meaning: Unchanging. It's a solid piece that won't shift. It won't go anywhere on you.

Feature

Definition: Part of an application that was planned to exist.

Meaning: A piece of the plan falling in place.

Open Source

Definition: Any program, where the actual source code has been given to the public to not only access, but to edit.

Meaning: Community Effort, open for commentary or change, change from outside sources.

If Then Statement

Definition: A system of logic checks, where the value of variables is checked against a set of expected values to determine the state.

Meaning: Actions have Consequences, for good or bad.