

INSTITUTO TECNOLÓGICO DE CHIHUAHUA II

Autor: José Sebastian López Ibarra

Docente: Rubén Alonso Hernández Chávez

25 / 08 / 2022

EDD – 09 – ISC

Índice

[Introducción 1](#_Toc112344790)

[Objetivos 2](#_Toc112344791)

[Texto 3](#_Toc112344792)

[Conclusiones 13](#_Toc112344793)

[Recomendaciones 14](#_Toc112344794)

[Referencias 15](#_Toc112344795)

# Introducción

La programación es un área que está en constante cambio. Los conceptos que surgieron en el pasado, han sido modificados y completamente reemplazados por nuevos conceptos.

El tener que estudiar, investigar, retener información en grandes cantidades, desencadena que el cerebro trabaje más de lo normal. El no saber para que funciona un código, la presión de terminar el código para antes del día establecido, genera una gran cantidad de estrés y ansiedad en el programador. Esto sólo provoca que se rinda, lo que incita a buscar alternativas “factibles” (trampas) para terminar con sus tareas.

# Objetivos

* Poner en práctica las habilidades y el conocimiento previo para poder traducir el texto de inglés a español, de una manera concisa artículos tanto expositivos como informativos sobre temas relacionados con la carrera profesional que estamos cursando actualmente.
* Al realizar esta tarea, el profesor tendrá una idea sobre el nivel de inglés que actualmente tenemos.

# Texto

INFORMACIÓN:

Se le proporciona el siguiente artículo en idioma inglés, el cual deberá traducir y darle el formato solicitado en el reglamento del curso (portada, índices, objetivo, introducción, cuerpo del documento, etc). Este documento tiene texto plano con diversas secciones para el uso adecuado de interlineados, títulos, formato de letra, etc. Incluye imágenes, las cuales deben ser incluidas en la tarea y correctamente etiquetadas e incluidas en el índice de figuras.

DATOS PARA LAS REFERENCIAS:

Web: http://mashable.com/2014/04/30/programming-sucks/#B8Ofmt7T0sq5

Autores: Peter Welch

Fecha y Editorial: Página Web(Mashable.com) 30 ABRIL 2014

Título: La programación no es un trabajo manual, pero aun así apesta.

INTRODUCCIÓN:

Cada amigo que tengo con un trabajo que implica recoger algo más pesado que una computadora portátil más de dos veces por semana, eventualmente encuentra una manera de incluir algo como esto en la conversación: "Hermano, no trabajas duro. Acabo de trabajar 4700 horas". semana cavando un túnel debajo de Mordor con un destornillador".

Tienen un punto. Mordor apesta, y ciertamente es más agotador físicamente cavar un túnel que tocar un teclado a menos que seas una hormiga. Pero, por el bien del argumento, ¿podemos estar de acuerdo en que el estrés y la locura son cosas malas? Asombroso. Bienvenido a la programación.

TODOS LOS EQUIPOS DE PROGRAMACIÓN ESTÁN CONSTRUIDOS POR Y DE GENTE LOCA

Imagina unirte a un equipo de ingenieros. Está entusiasmado y lleno de ideas, probablemente recién egresado de la escuela y un mundo de diseños limpios y hermosos, impresionantes en su unidad estética de propósito, economía y fuerza. Comienza conociendo a Mary, líder del proyecto de un puente en un área metropolitana importante. Mary te presenta a Fred, después de pasar los quince controles de seguridad instalados por Dave porque a Dave le robaron el suéter de su escritorio una vez y nunca más. Fred solo trabaja con madera, por lo que le preguntas por qué está involucrado porque se supone que este puente debe permitir que el tráfico de la hora pico lleno de automóviles llenos de humanos mortales cruce una caída de 200 pies. No te preocupes, dice Mary, Fred se encargará de los pasillos. ¿Qué pasarelas? Bueno, Fred hizo un buen caso para las pasarelas y se sumarán al atractivo del puente. Por supuesto, tendrán que construirse sin barandillas, porque Phil, que no es ingeniero, aplica una regla estricta de no barandillas. Nadie está seguro de lo que hace Phil, pero definitivamente está lleno de sinergia y tiene que ver con la alta gerencia, con quien ninguno de los ingenieros quiere tratar, así que simplemente dejan que Phil haga lo que él quiere.

Sara, mientras tanto, encontró varias técnicas de pavimentación de bordes hemorrágicos y las incorporó al diseño del puente, por lo que tendrá que construir alrededor de cada una a medida que progresa el puente, ya que cada una significa diferentes problemas subyacentes de soporte y seguridad. Tom y Harry han estado trabajando juntos por años, pero tienen una disputa constante sobre si usar medidas métricas o imperiales, y se ha convertido en un caso de "quien llegó primero a esa parte del diseño". Esto ha sido un dolor de cabeza para las personas que realmente atornillan las cosas, se dieron por vencidas y simplemente forzaron, martillaron o soldaron su camino durante el día con las piezas que tenían a mano. Además, el puente fue diseñado como un puente colgante, pero en realidad nadie sabía cómo construir un puente colgante, por lo que llegaron a la mitad y luego simplemente agregaron columnas de soporte adicionales para mantenerlo en pie, pero dejaron los cables de suspensión porque todavía estoy sosteniendo partes del puente. Nadie sabe qué partes, pero todos están bastante seguros de que son partes importantes. Después de hacer las presentaciones, se le invita a presentar algunas ideas nuevas, pero no tiene ninguna porque es ingeniero de propulsión y no sabe nada sobre puentes.

¿Cruzarías este puente? No. Si de alguna manera se construyera, todos los involucrados serían ejecutados. Sin embargo, alguna versión de esta dinámica escribió todos y cada uno de los programas que usó alguna vez, software bancario, sitios web y un programa de uso generalizado que se suponía que protegería la información en Internet, pero no lo hizo.



ERROR CATASTRÓFICO

El usuario intentó usar el programa de la manera en que el programa estaba destinado a ser usado

Opciones:

1. Borrar computadora
2. Llorar

1. IMAGEN: PETER WELCH

TODO EL CÓDIGO ES MALO

Todos los programadores ocasionalmente, cuando no hay nadie en casa, apagan las luces, se sirven un vaso de whisky escocés, ponen música electrónica ligera en alemán y abren un archivo en su computadora. Es un archivo diferente para cada programador. A veces lo escribieron, a veces lo encontraron y supieron que tenían que salvarlo. Leen las líneas y lloran por su belleza, luego las lágrimas se vuelven amargas al recordar el resto de los archivos y el inevitable colapso de todo lo que es bueno y verdadero en el mundo.

Este archivo es buen código. Tiene nombres sensatos y consistentes para funciones y variables. Es conciso. No hace nada obviamente estúpido. Nunca ha tenido que vivir en la naturaleza o responder a un equipo de ventas. Hace exactamente una cosa mundana y específica, y lo hace bien. Fue escrito por una sola persona, y nunca tocado por otra. Se lee como poesía escrita por alguien mayor de treinta años.

Todo programador comienza escribiendo un pequeño copo de nieve perfecto como este. Luego se les dice el viernes que necesitan tener seiscientos copos de nieve escritos para el martes, por lo que hacen un poco de trampa aquí y allá y tal vez copian algunos copos de nieve y tratan de unirlos o tienen que pedirle a un compañero de trabajo que trabaje en uno que lo derrite y luego todos los copos de nieve de los programadores se juntan en una forma inescrutable y alguien apoya un Picasso en él porque nadie quiere ver la orina de gato empapando todos tus copos de nieve rotos derritiéndose a la luz del día. La próxima semana, todo el mundo echa más nieve para evitar que el Picasso se caiga.

Existe la teoría de que puede curar esto siguiendo los estándares, excepto que hay más "estándares" que cosas que las computadoras realmente pueden hacer, y estos estándares se mejoran y difaman de diversas formas según las preferencias personales de las personas que los codifican, por lo que no hay colección de código que haya llegado alguna vez al mundo real sin hacer unas pocas docenas de cosas idénticas, unas pocas docenas de formas ni remotamente similares. Las primeras semanas de cualquier trabajo consisten simplemente en descubrir cómo funciona un programa, incluso si está familiarizado con todos los lenguajes, marcos y estándares involucrados, porque los estándares son unicornios.

SIEMPRE HABRÁ OSCURIDAD

Pasé algunos años creciendo con un armario en mi dormitorio. El armario tenía un diseño extraño. Al principio se veía normal, luego entraste para hacer las cosas del armario y descubriste que la pared a tu derecha daba paso a una alcoba, convirtiéndose en un pequeño estante útil. Luego miró hacia arriba, y la pared en la parte trasera de la alcoba daba paso de nuevo, en un espacio angosto de absoluta nada, donde ninguna luz podía caer y que inmediatamente identificó como el refugio diurno para cada monstruo voraz que mantuvo a raya con linternas y animales de peluche cada noche.

Esto es lo que es aprender a programar. Llegas a conocer tus herramientas útiles, luego miras a tu alrededor, y hay algunas herramientas nuevas y útiles cerca y esas herramientas te muestran el horror sin fondo que siempre estuvo justo al lado de tu cama.

Por ejemplo, supongamos que eres un desarrollador web promedio. Estás familiarizado con una docena de lenguajes de programación, toneladas de bibliotecas útiles, estándares, protocolos, lo que tienes. Todavía tiene que aprender más a un ritmo de aproximadamente uno por semana, y recuerde verificar los cientos de cosas que sabe para ver si se han actualizado o roto y asegurarse de que todos siguen funcionando juntos y que nadie solucionó el error, uno de ellos que explotaste para hacer algo que pensaste que era realmente inteligente un fin de semana cuando estabas borracho. Estás al día, así que está bien, luego todo se quiebra.

"¿Doble tee eff?" dices, y comienzas a buscar el problema. Descubres que un día, un idiota decidió que, dado que otro idiota decidió que 1/0 debería ser igual a infinito, podrían usar eso como una abreviatura de "Infinito" al simplificar su código. Luego, un no idiota decidió correctamente que esto era una idiotez, que es lo que el idiota original debería haber decidido, pero como no lo hizo, el no idiota decidió ser un imbécil y convertir esto en un error fallido en su nuevo compilador. Luego decidió que no iba a decirle a nadie que esto era un error, porque él es un imbécil, y ahora todos tus copos de nieve son orina y ni siquiera puedes encontrar al gato.

Eres un experto en todas estas tecnologías, y eso es bueno, porque esa experiencia te permitió dedicar solo seis horas a descubrir qué salió mal, en lugar de perder tu trabajo. Ahora tienes un pequeño dato extra para guardar entre los millones de pequeños datos que tienes que memorizar porque muchos de los programas de los que dependes están escritos por imbéciles e idiotas.

Y eso es solo en su propio campo elegido, que representa una fracción tan pequeña de todas las cosas que hay que saber en informática que es mejor que nunca haya aprendido nada en absoluto. Ni una sola persona viva sabe cómo funciona realmente todo en su MacBook de cinco años. ¿Por qué te decimos que lo apagues y vuelvas a encenderlo? Porque no tenemos la menor idea de qué es lo que está mal, y es realmente fácil inducir al coma en las computadoras y hacer que su equipo integrado de médicos automáticos intente resolverlo por nosotros. La única razón por la que las computadoras de los codificadores funcionan mejor que las de los no codificadores es que los codificadores saben que las computadoras son niños pequeños esquizofrénicos con enfermedades autoinmunes y no los derrotamos cuando son malos.

SE HA HECHO MUCHO TRABAJO EN INTERNET Y EL INTERNET ES SU PROPIO INFIERNO ESPECIAL

¿Recuerdas esas cosas sobre gente loca y código malo? Internet es eso, excepto que es literalmente mil millones de veces peor. Los sitios web que son carros de compras glorificados con quizás tres páginas dinámicas son mantenidos por equipos de personas las 24 horas del día, porque la verdad es que todo se rompe todo el tiempo, en todas partes, para todos. En este instante, alguien que trabaja para Facebook recibe decenas de miles de mensajes de error y trata frenéticamente de encontrar el problema antes de que se derrumbe toda la farsa. Hay un equipo en una oficina de Google que no ha dormido en tres días. En algún lugar hay una programadora de bases de datos rodeada de botellas vacías de Mountain Dew cuyo marido cree que está muerta. Y si esta gente se detiene, el mundo arde. La mayoría de la gente ni siquiera sabe lo que hacen los administradores de sistemas, pero créeme, si todos tomaran un descanso para almorzar al mismo tiempo, no llegarían a la tienda de delicatessen antes de que te quedaras sin balas para proteger tus productos enlatados de bandas itinerantes de mutantes.

No puedes reiniciar Internet. Billones de dólares dependen de una telaraña desvencijada de acuerdos no oficiales y código "suficientemente bueno por ahora" con comentarios como "QUE HACER: ARREGLAR ESTO ES UN HACK REALMENTE PELIGROSO, PERO NO SÉ QUÉ PASA" que se escribieron hace diez años. Ni siquiera he mencionado las legiones de personas que atacan varias partes de Internet por espionaje y ganancias o porque están aburridas. ¿Has oído hablar de 4chan? 4chan podría destruir tu vida y tu negocio porque decidieron que no les gustabas por una tarde, y ni siquiera nos preocupamos por 4chan porque otra bomba nuclear no hace mucha diferencia en un invierno nuclear.

En Internet, está bien decir: "Sabes, esto funciona algunas veces si estás usando la tecnología adecuada", y ¡BAM! ahora es parte de Internet. Cualquiera con un par de cientos de dólares y una computadora puede enganchar un poco de Internet y colocar cualquier horrible fragmento de código de pirateo que quiera y luego adjuntar su pequeño fragmento a un montón de fragmentos grandes y todo se pone un poco peor. Incluso los buenos programadores no se molestan en aprender las especificaciones arcanas descritas por las organizaciones que la gente creó para implementar algunos unicornios, por lo que todos pasan la mitad de su tiempo lidiando con el hecho de que nada coincide con nada o tiene algún sentido y podría fallar en cualquier momento y solo tratamos de ocultarlo y esperamos que nadie se dé cuenta.

Estas son las reglas secretas de Internet: cinco minutos después de abrir un navegador web por primera vez, un niño en Rusia tiene su número de seguro social. ¿Te apuntaste a algo? Una computadora en la NSA ahora rastrea automáticamente su ubicación física por el resto de su vida. ¿Enviado un correo electrónico? Su dirección de correo electrónico acaba de aparecer en una cartelera en Nigeria.

Estas cosas no son ciertas porque no nos importa y no tratamos de detenerlas, son ciertas porque todo está roto porque no hay un buen código y todos intentan que siga funcionando. Ese es tu trabajo si trabajas con Internet: esperar que lo último que escribiste sea lo suficientemente bueno como para sobrevivir unas horas para que puedas cenar y tomar una siesta.

NO EMPEZAMOS LOCOS, NOS ESTÁN VOLVIENDO LOCOS



ERROR: Intentó analizar HTML con expresión regular, el sistema devolvió Cthulhu

2. ERRORES

Gracioso, ¿verdad? ¿No? Que tal este intercambio:

"¿Eso se llama arrayReverse?"

"s/camello/\_/"

"Genial, gracias."

¿No fue útil ese tipo? ¿Con el camello? ¿No te parece una respuesta apropiada? ¿No? Bien. Todavía puedes encontrar a Jesús. Aún no has pasado tanto tiempo de tu vida leyendo código como para empezar a hablar en él. El cerebro humano no es particularmente bueno en lógica básica y ahora hay toda una carrera en hacer nada más que lógica muy, muy compleja. Se deben seleccionar grandes cadenas de condiciones y requisitos abstractos para descubrir cosas como comas faltantes. Hacer esto todo el día te deja en un estado de afasia leve mientras miras las caras de las personas mientras hablan y no sabes si han terminado porque no hay punto y coma. Te sumerges en un mundo de total falta de sentido donde lo único que importa es una pequeña serie de números que entró en un laberinto gigante de símbolos y una serie diferente de números o la imagen de un gatito salió por el otro extremo.

El impacto destructivo en el cerebro está demostrado por los lenguajes de programación que escribe la gente. Este es un programa:



3. PROGRAMA 1

Ese programa hace exactamente lo mismo que este programa:



4. PROGRAMA 2

Y este programa:



5. PROGRAMA 3

Y este otro:



6. PROGRAMA 4

Y una vez alguien escribió un lenguaje de programación que permitía que alguien más escribiera esto:



7. PROGRAMA 5

Según el autor, ese programa consta de "dos líneas de código que analizan dos líneas de comentarios incrustados en el código para leer los números mayas que representan los caracteres ASCII individuales que componen el título de la revista, representados en arte ASCII rotado en 90 grados".

Ese programa ganó un concurso, porque claro que sí. ¿Quieres vivir en un mundo así? No. Este es un mundo en el que puedes fumar un paquete al día y nadie lo cuestiona. "Por supuesto que fuma un paquete al día, ¿quién no lo haría?" Eventualmente, todos los programadores se despiertan y, antes de que estén completamente conscientes, ven su mundo entero y cada relación en él como fragmentos de código, e intercambian historias al respecto como si la somnolencia desencadenante de viajes con ácido fuera algo normal que le sucede a la gente. Este es un mundo donde la gente evita el sexo para escribir un lenguaje de programación para orangutanes. Todos los programadores están obligando a sus cerebros a hacer cosas que los cerebros nunca debían hacer en una situación que nunca pueden mejorar, de diez a quince horas al día, de cinco a siete días a la semana, y cada uno de ellos se está volviendo loco lentamente.

</rant>

Así que no, no estoy obligado a poder levantar objetos que pesen hasta cincuenta libras. Cambié eso por la oportunidad de cortarle el vello púbico a Satanás mientras él cena en mi cráneo abierto, así que algunas partes de Internet seguirán funcionando durante unos días más.

# Conclusiones

Ningún programador, ni los que tienen altos puestos están exentos de que muchas veces leer un código, entenderlo y saber interpretarlo no es tan simple como parece. En la mayoría de los casos, tratar de resolver un problema cuando se trata de un código, no es algo que suene bonito, al contrario, al escuchar “error” entramos en un momento de crisis.

Los programadores son personas que viven en constante trabajo, en su vocabulario no existen los descansos. Siempre están resolviendo cuestiones como el por qué no funciona un código, en donde está el error, entre otros.

# Recomendaciones

- Que los alumnos pasen a realizar prácticas de los temas vistos en las diapositivas para romper ese miedo de no saber qué hacer.

# Referencias

* Wickware, C. (2014, April 30). *Programming Isn’t Manual Labor, but it still sucks*. Mashable. <http://mashable.com/2014/04/30/programming-sucks/>