

TECNOLOGICO NACIONAL DE MEXICO
INSTITUTO TECNOLÓGICO DE CHIHUAHUA II



DEPARTAMENTO DE SISTEMAS Y COMPUTACION
INGENIERIA EN SISTEMAS COMPUTACIONALES
ESTRUCTURAS DE DATOS
TAREA #1 "TRADUCCION Y FORMATO"
PRIMERA EVALUACION PARCIAL

IVAN EDUARDO CASTILLO CASTRO
20550390

ÍNDICE

Introducción.....	3
Objetivo.....	3
La Programación no es un Labor Manual, pero aun así Apesta.....	3
Todos los equipos de Programación son contruidos por una persona loca.....	4-5
Todo Código es Malo.....	5-6
Siempre Habrá Oscuridad.....	7-8
Se Hace Mucho Trabajo en Internet y el Internet es su Propio Paisaje Especial.....	8-9
No Empezamos Locos, nos Estamos Volviendo Locos.....	10-13
Conclusión.....	13
Recomendaciones.....	13-14
Referencias.....	13

INDICE DE IMÁGENES

Imagen 1. Peter Welch.....	5
Imagen 2. ERROR.....	10
Imagen 3. Programa 1.....	11
Imagen 4. Programa 2.....	11
Imagen 5. Programa 3.....	11
Imagen 6. Programa 4.....	12
Imagen 7. Programa 5.....	12

INTRODUCCION

Mi introducción es que con este trabajo adquirir un poco de conocimiento del inglés ya que es una lengua muy importante el inglés ya que en la mayoría de lenguajes programables es muy importante el conocimiento del inglés.

Hablando fuera del tema escolar, me servirá en el ambiente laboral consiguiendo un trabajo dedicado a sistemas, trabajando como desarrollador.

OBJETIVO.

El objetivo de este trabajo es familiarizarnos en el Inglés ya que en la programación solemos llevarlo en varias variables o palabras reservadas.

También nos servirá para ir practicando para que mientras programemos ya las palabras nos acordemos de como se escriben sin necesidad de usar el traductor.

LA PROGRAMACION NO ES UNA LABOR MANUAL PERO AUN ASI APESTA

Cada amigo tiene un trabajo que implica levantar cosas más pesadas que una laptop más de dos veces a la, eventualmente encuentra una manera de deslizar cosas como esta conversación: "Hermano, tu no trabajas duro. Yo recién trabaje 4700 horas cavando un túnel debajo de Mordor con un desarmador".

Ellos tienen un punto. Mordor apesta, y ciertamente eso es más agotador que tocar un teclado como si fueras una hormiga. Pero por el bien del argumento, ¿podemos nosotros agregar que el estrés y la locura son cosas malas? Impresionante. Bienvenidos a la Programación.

TODOS LOS EQUIPOS DE PROGRAMACION ESTAN HECHOS POR GENTE LOCA

Imaginen juntarse a un equipo de ingenieros. Estuvieras emocionado y lleno de ideas, probablemente recién salido de escuela y un mundo de limpieza, hermosos diseños, inspiradores en su unidad de propósito, economía y fuerza. Tú empiezas conociendo a Mary, una líder del proyecto de un puente para 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 su suéter de escritorio una vez más. Fred solo trabaja con madera así que preguntas ¿Por qué está Involucrado? Porque se supone que este puente permite que el tráfico de la hora apunta lleno de autos, llenos de pasos personas cruzando una caída de 200 pies sobre los rápidos.

No te preocupes, dice Mary, Fred va a manejar las pasarelas. ¿Qué pasarelas? Bueno Fred hizo un buen caso para las pasarelas y sumara al atractivo. Por supuesto, tendrán que construirse sin barandillas, porque Phil, que no es ingeniero, aplica una estricta regla de no barandillas.

Nadie está seguro de lo que hace Phil, pero definitivamente está lleno de energía y tiene que ver con la alta dirección, con quien ningún ingeniero quiere tratos, así que simplemente dejan que Phil haga lo que quiera.

Sara, mientras, tanto, ha encontrado varias técnicas de pavimentación con bordes de hemorragia y las ha trabajado todas alrededor del diseño del puente, por lo que tendrá que construir alrededor de cada una a medida que avanza el puente, ya que cada una significa diferentes preocupaciones de seguridad y soporte subyacentes. Tom y Harry han estado trabajando juntos durante años, pero tienen una disputa constante sobre si usar medidas métricas o imperiales, y se ha convertido en un caso de “Quien llego a esa parte del diseño”. Esto ha sido un dolor de cabeza para la gente que realmente está atornillando cosas, se han rendido y simplemente han forzado, martillando o soldando su camino a lo largo del día con las partes que tenían en la mano. Además el puente ha sido diseñado como un puente

colgante, pero nadie sabía cómo construir un puente colgante, por lo que llegaron a la mitad y luego simplemente agregaron columnas de soporte adicionales para mantener el puente en pie, pero dejaron los cables de suspensión porque “Todavía está sosteniendo partes del puente”. Nadie sabe que partes, pero todos están bastantes seguros de que son partes importantes. Una vez realizadas las presentaciones, se les invita a proponer algunas ideas nuevas, pero no tienen ninguna porque son ingenieros en propulsión y no sabe nada sobre puentes. ¿Cruzarías este puente? No. Si de alguna manera se construyera todos los involucrados serian ejecutados. Sin embargo, alguna versión de esta dinámica escribió todos los programas que ha utilizado, software bancario, sitios web y un programa omnipresente que se suponía que protegía la información en internet, pero no lo hizo.

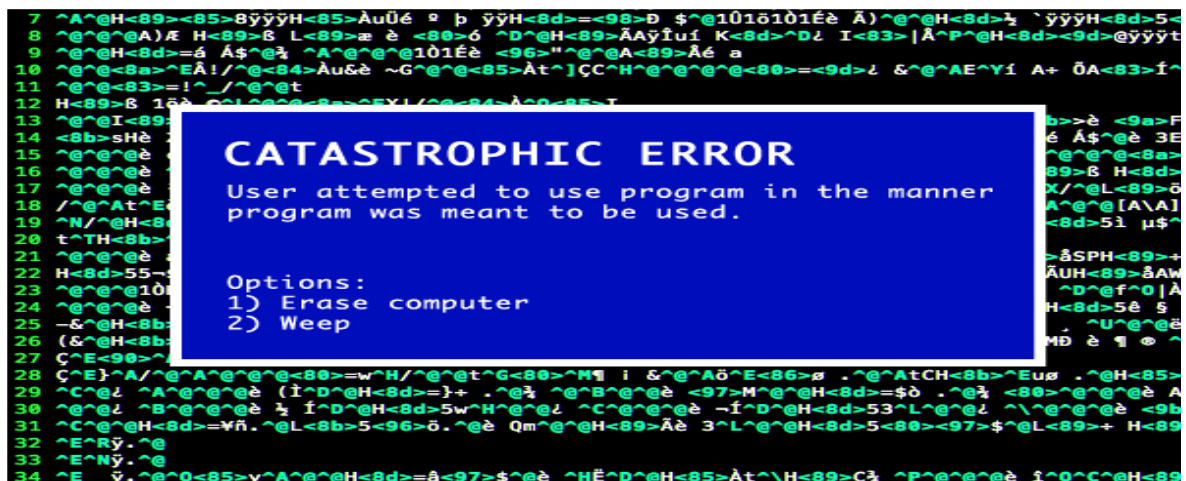


Imagen 1. Peter Welch

TODO CODIGO ES MALO

Todos los programadores de vez en cuando, cuando no hay nadie en casa, apagan las luces, se sirven un vaso de whisky, ponen algo de música electrónica alemana ligera 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 guardarlo. 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 bueno y verdadero del mundo.

Este archivo es buen código. Tiene nombres sensibles 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 fue tocado por otra. Parece poesía escrita por alguien mayor de treinta años.

Todo programador empieza escribiendo un pequeño copo de nieve perfecto como este. Luego, el viernes les dicen que necesitan tener seiscientos copos de nieve escritos para el martes, así que hacen un poco de trampa aquí y allá y tal vez copian algunos copos de nieve e intentan pegarlos 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 sobre él porque nadie quiere ver la orina de gato empapando todos tus copos de nieve rotos derritiéndose con la luz del día. La semana que viene, todo el mundo le echa más nieve para evitar que el Picasso se caiga.

Existe la teoría de que puede curar esto siguiendo estándares, excepto que hay más "estándares" que cosas que las computadoras pueden hacer realmente, y estos estándares se mejoran y difaman de diversas maneras según las preferencias personales de las personas que los codifican, por lo que no hay recopilación de código ha llegado 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 son simplemente descubrir cómo funciona un programa, incluso si está familiarizado con todos los lenguajes, marcos y estándares involucrados, porque los estándares son unicornios.

SIEMPRE HABRA OSCURIDAD

Pasé algunos años creciendo con un armario en mi habitación. El armario tenía un diseño extraño. Al principio parecía normal, luego entraste para hacer las cosas del armario y descubriste que la pared de tu derecha daba paso a una alcoba, lo que lo convertía en un pequeño estante útil. Luego miraste hacia arriba, y la pared en la parte trasera de la alcoba cedió de nuevo, en un espacio de acceso de la nada absoluta, donde no podía caer la luz y que inmediatamente identificaste como el refugio diurno para cada monstruo voraz que mantuviste 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 al lado de tu cama.

Por ejemplo, supongamos que es un desarrollador web medio. Está familiarizado con una docena de lenguajes de programación, toneladas de útiles bibliotecas, estándares, protocolos, lo que sea. 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 todavía funcionen juntos y que nadie haya solucionado el error en uno de ellos que aprovechaste para hacer algo que pensaste que era realmente inteligente un fin de semana cuando estabas borracho. Estás actualizado, así que está bien, luego todo se rompe.

"¿Doble tu tee eff?" dices, y empieza a buscar el problema. Descubres que un día, algún 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. Entonces, un no idiota decidió acertadamente que esto era idiota, que es lo que el idiota original debería haber decidido, pero como no lo hizo, el no idiota decidió ser un idiota y convertir esto en un error fallido en su nuevo compilador. Luego decidió que

no le iba a decir a nadie que esto era un error, porque es un idiota, y ahora todos tus copos de nieve son orina y ni siquiera puedes encontrar al gato.

Usted es un experto en todas estas tecnologías, y eso es bueno, porque esa experiencia le permite dedicar solo seis horas a averiguar qué salió mal, en lugar de perder su trabajo. Ahora tiene un pequeño dato adicional para guardar entre los millones de pequeños hechos que tiene que memorizar porque muchos de los programas de los que depende están escritos por 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 posible 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 le decimos que lo apague y vuelva a encender? Porque no tenemos la menor idea de lo que está mal, y es realmente fácil inducir el 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 programadores funcionan mejor que las computadoras de los que no lo son es que los programadores saben que las computadoras son niños pequeños esquizofrénicos con enfermedades autoinmunes y no los golpeamos cuando son malos.

SE HACE MUCHO TRABAJO EN INTERNET Y EL INTERNET ES SU PROPIO PAISAJE ESPECIAL

¿Recuerdas esas cosas sobre los locos y el código incorrecto? Internet es eso, excepto que es literalmente mil millones de veces peor. Los sitios web que son carritos de compras glorificados con quizás tres páginas dinámicas son mantenidos por equipos de personas durante todo el día, porque la verdad es que todo se está rompiendo todo el tiempo, en todas partes, para todos. En este momento, alguien que trabaja para Facebook está recibiendo decenas de miles de mensajes de error y está tratando

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 un programador de bases de datos rodeado 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 créame, si todos se tomaran un descanso para almorzar al mismo tiempo, no llegarían a la tienda de delicades antes de que usted se quedara sin balas para proteger sus productos enlatados de las bandas errantes de mutantes. .

No puedes reiniciar Internet. Trillones de dólares dependen de una desvencijada telaraña de acuerdos no oficiales y de un código "suficientemente bueno por ahora" con comentarios como "TODO: ARREGLAR ESTO ES UN HACK REALMENTE PELIGROSO PERO NO SÉ LO QUE ESTÁ MAL" que se escribieron hace diez años. Ni siquiera he mencionado las legiones de personas que atacan varias partes de Internet por espionaje y lucro o porque están aburridas. ¿Has oído hablar de 4chan? 4chan podría destruir tu vida y tu negocio porque decidieron que no les agradaste 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 fragmento de código de pirateo que quiera y luego adjuntar su parte a un montón de partes 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 sentido y podría romperse en cualquier momento y solo tratamos de encubrirlo y esperamos que nadie se dé cuenta.

NO EMPEZAMOS LOCOS, NOS ESTAMOS VOLVIENDO LOCOS

| ERROR: Attempted to parse HTML with regular expression; system returned Cthulhu.

Imagen 2. Error

¿Gracioso, verdad? ¿No? ¿Qué tal este intercambio?

"¿Eso se llama arrayReverse?"

"s / camello / _ /"

"Genial, gracias."

¿No fue útil ese tipo? ¿Con el camello? ¿No parece una respuesta apropiada? ¿No? Bueno. Todavía puedes encontrar a Jesús. Todavía no ha pasado tanto tiempo leyendo el 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 realmente compleja. Se deben seleccionar grandes cadenas de condiciones y requisitos abstractos para descubrir cosas como la falta de comas. 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 que han terminado porque no hay punto y coma. Te sumerges en un mundo de total sinsentido donde lo único que importa es una pequeña serie de números que entraron en un laberinto gigante de símbolos y una serie diferente de números o una imagen de un gatito salió por el otro extremo.

El impacto destructivo en el cerebro lo demuestran los lenguajes de programación que la gente escribe.

Este es un programa:

```
#include
int main( int argc, char** argv ) {
std::cout return 0;
}
```

Imagen 3. Programa 1

Ese programa hace exactamente lo mismo que este programa:

```
`r'~~~~~.H.e.l.l.o. .w.o.r.l.d.i
```

Imagen 4. Programa 2

Y este programa:

```
>++++++[<++++++>-]++++++[<++++>-]<+.+++++..+++. [-] >++++++
[<++++>-] +++++++[<++++++>-]<-._____.+++ ._____.
[-]>++++++[<++++>- ]<+. [-]++++++.
```

Imagen 5. Programa 3

Y éste:

```

Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook.
Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook? Ook! Ook! Ook? Ook! Ook?
Ook.
Ook! Ook. Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook.
Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook?
Ook! Ook! Ook? Ook! Ook? Ook. Ook. Ook. Ook! Ook. Ook. Ook. Ook. Ook. Ook.
Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook. Ook! Ook. Ook. Ook. Ook.
Ook. Ook. Ook! Ook. Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook. Ook.
Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook.
Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook? Ook! Ook! Ook? Ook! Ook? Ook. Ook!
Ook.
Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook.
Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook? Ook! Ook! Ook? Ook! Ook? Ook. Ook! Ook! Ook! Ook! Ook! Ook! Ook!
Ook.
Ook? Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook! Ook. Ook. Ook. Ook. Ook. Ook.
Ook.
Ook! Ook. Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook.
Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook!
Ook! Ook. Ook. Ook? Ook. Ook? Ook. Ook. Ook! Ook. Ook! Ook? Ook! Ook! Ook?
Ook!
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook! Ook.

```

Imagen 6. Programa 4

Y una vez que alguien escribió un lenguaje de programación que le permitió a otra persona escribir esto:

```

#:: :-| :-| :-| :-| 0-| :-| :-| :-| :-| :-|
open(Q,$0);while("){if(/^#(.*)$/){for(split("-", $1)){ $q=0;for(split){s/|
/::/xg;s/::/g;$Q=$_?length:$_;$q+=$q?$Q:$Q*20;}print chr($q);}}print"n";
#:: :-| :-| :-| :-| :-| :-| :-| :-| :-| :-|
"

```

Imagen 7. Programa 5

Según el autor, ese programa es "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 90 grados".

Ese programa ganó un concurso, porque por supuesto que lo hizo.

¿Quieres vivir en un mundo como este? 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?" Con el tiempo, todos los

programadores se despiertan y, antes de estar completamente conscientes, ven todo su mundo y cada relación que hay en él como fragmentos de código, e intercambian historias al respecto como si la somnolencia que desencadena viajes con ácido es 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 forzando a sus cerebros a hacer cosas que los cerebros nunca debieron hacer en una situación que nunca podrán 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 veinticinco kilos. Cambié eso por la oportunidad de recortar el vello púbico de Satanás mientras cena fuera de mi cráneo abierto para que algunos fragmentos de Internet continúen funcionando durante unos días más.

CONCLUSIONES

Al finalizar el trabajo me di cuenta que es muy necesario poder practicar el Inglés ya que al momento de programar y de tener errores en NetBeans podremos descifrar cual es nuestro error de una manera más rápida y nosotros poder saber cuál es la palabra correcta sin necesidad de resolverlo por medio de las sugerencias que nos da un lenguaje de programación o un sistema programable.

RECOMENDACIONES

Mis recomendaciones son que tratemos de descifrar las palabras en inglés, no aprender en su totalidad sino que como apenas nos familiarizamos en el inglés, derivado de la programación tratar de saber cuáles son nuestros

errores, a lo mejor no sabremos lo que dice en si el problema sino que traducir propiamente uno cual es el problema y saber cuál será la solución.

REFERENCIAS BIBLIOGRÁFICAS.

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

Autores: Peter Welch

Fecha y Editorial: Pagina Web (Mashable.com) Abril 30, 2014

Título: La Programación no es una Labor Manual, Pero aun así Apesta