

ANEXO 1

| MI PROYECTO DE VIDA | | | | |
|---------------------|-------------------------|---|--------------------------------------|---|
| ÁMBITO | OBJETIVO ¿Qué deseo? | TIEMPO ¿En cuánto tiempo lo lograré? | ESTRATEGIAS ¿Cómo le voy a hacer? | APOYOS EXTERNOS ¿En quién me puedo apoyar para lograrlo? |
| PERSONAL | | | | |
| ESCOLAR | | | | |
| FAMILIAR | | | | |
| LABORAL | | | | |

ANEXO 2
ACTITUDES POSITIVAS

| | | |
|----------------|--------------|------------|
| MISERICORDIOSO | COOPERADOR | ALEGRE |
| DECIDIDO | SEGURO | SENCILLO |
| EXPRESIVO | REFLEXIVO | OPTIMISTA |
| TOLERANTE | ESPONTANEO | CONSTANTE |
| ATENTO | ACOGEDOR | SENSIBLE |
| SOLIDARIO | CÁLIDO | COMPENSIVO |
| LUCHADOR | RESPETUOSO | FUERTE |
| ENTUSIASTA | GENEROSO | ESFORZADO |
| ORDENADO | IDEALISTA | CREATIVO |
| JUSTO | PERSEVERANTE | OBSERVADOR |
| LIBRE | VALIENTE | PROFUNDO |
| EMPRENDEDOR | LEAL | CARIÑOSO |
| RESPONSABLE | AMISTOSO | SERVICIAL |
| HONESTO | BONDADOSO | CAPAZ |
| INTELIGENTE | AGRADABLE | LIMPIO |
| TRABAJADOR | NOBLE | CORTÉS |
| CURIOSO | RECEPTIVO | AGRADECIDO |

ESTAS SON MIS ACTITUDES:

ESTAS ACTITUDES DESEO DESARROLLAR:

ANEXO 3

ORGANIZACIÓN PARA EL ESTUDIO

Los factores ambientales son aquellos elementos externos del medio ambiente que inciden favorable o desfavorablemente en la calidad del estudio realizado por el alumno. La organización para el estudio es la disposición ordenada de los elementos que componen al acto de estudiar. Los elementos de la organización son: *lugar, mente, tiempo*.

A) Organizar el lugar:

Se refiere a las condiciones físicas del espacio y al lugar de estudio. Se debe estar libre de distractores, silencioso, solitario y bien iluminado, de temperatura agradable y alejada de ruidos e interrupciones. La ventilación y renovación frecuente del aire del lugar de trabajo favorecen la oxigenación del cerebro y aumentan la atención. Considerar una mesa de trabajo con todo el material necesario a mano a fin de no dispersar energías.

B) Organizar la mente:

Significa tener una idea general, básica del tema para luego analizar y deducir a fondo las partes que lo componen. Irse de lo general a lo particular. Los contenidos adquieren significado cuando se descubre la relación entre todos ellos.

C) Organizar el tiempo:

Significa adaptarlo tanto al trabajo que se va a realizar como a las características propias de cada persona. Saber analizar qué tipos de tareas pueden hacerse en el último momento y cuáles hay que realizar a lo largo del tiempo es un factor relevante en el estudio, y por lo tanto, de un buen rendimiento académico. Facilita la concentración, al crear el hábito de estudiar determinadas materias en un instante y lugar determinado y permite aprovechar el tiempo libre para la recreación.

El tiempo que se necesita para cada asignatura no es igual, dependerá de la "aptitud" para el ramo, su conocimiento previo, su interés y la dificultad de la materia misma.

En la planificación del tiempo se deberán considerar los periodos libres dedicados al descanso, la diversión, y las obligaciones familiares y sociales impostergables. El tiempo libre bien empleado permitirá desarrollar los talentos, el encuentro social y afectivo.

Ventajas de una adecuada planificación:

- ✓ Hace posible distribuir equilibradamente la jornada diaria o semanal, teniendo en consideración las necesidades particulares de cada uno.
- ✓ Permite disponer de tiempo suficiente para completar todo el trabajo.
- ✓ Asegura un aprendizaje efectivo.
- ✓ Facilita la concentración.
- ✓ Permite sentirse seguro, obtener mayor confianza en sí mismo, superar el nerviosismo causante de la desorganización y sobrecarga de trabajo.
- ✓ Evita los sentimientos de culpa que resultan de no estudiar y permite disfrutar ampliamente del resto de tiempo de ocio.

El éxito de la distribución del tiempo dependerá del cuidado con el que se planifique. Debe hacerse de manera realista y objetiva, no idealizar planificando horarios con ninguna posibilidad de cumplir. Considere: horario de clases, necesidades de preparación para cada asignatura, periodos de descanso, horas de comida, sueño y especialmente los límites de la habilidad y destreza para alcanzar el dominio de las materias de estudio.

ESTRATEGIAS PARA DESARROLLAR LOS FACTORES AMBIENTALES Y ORGANIZACIÓN DEL ESTUDIO

1. Favorece la organización del lugar.

Es aconsejable disponer de un lugar determinado para estudiar, esto no quiere decir que no se pueda estudiar en otro lugar y con otras condiciones. Es necesario tomar en cuenta los factores externos que condicionan positivamente el estudio, tales como:

- Ordenar el lugar de trabajo, desechando lo que no es necesario para aprender. El lugar deberá estar libre de distractores (TV, radio, fotos, personas, ruidos molestos).
- Tener a la mano todo el material necesario (lápices, cuadernos, libros, diccionario, etc.).
- Temperatura adecuada. Ventilación adecuada.
- Buena iluminación.
- Actitud mental correcta: deseo de estudiar.
- No estudiar en cama acostado, la cama es un lugar asociado al descanso.

1.2. Posiciones adecuadas para el estudio.

- Columna vertebral recta. Cuerpo ocupando todo el espacio de la silla.
- Cabeza, ojos y oídos de frente al material de estudio o del trabajo escolar.
- Respirar lenta y profundamente.
- Concentrarse en el estado de relajación y descanso.

2. Estrategias para estimular la organización del pensamiento.

- Conoce la organización del texto: introducción, desarrollo, conclusiones.
- Clasifica la información a través de esquemas y resúmenes.
- Expresa en forma verbal lo aprendido.
- Escribe tu interpretación de lo que has estudiado.

3. Estrategias para la organización del tiempo

- Trabaja sistemáticamente para no verte abrumado por el exceso de trabajo a última hora.
- Distribuye tu tiempo en función de las actividades a realizar.
- Programa las horas de estudio de acuerdo a las exigencias y dificultad de cada materia.
- Planifica el tiempo libre para poder disfrutarlo después de que hayas cumplido tus obligaciones.
- Deja un tiempo para el repaso y la revisión de materias. Estudia todos los días una misma cantidad de tiempo, sitúa las materias difíciles, para cuando estés más descansado.
- Estudia cinco días a la semana, y deja el domingo para descansar.
- Empieza a estudiar por algo que te sea fácil o que te guste.
- Es mejor distribuir el tiempo entre varias actividades que dedicar todo el tiempo a una sola. Te cansas menos y rindes más.

El aprendizaje se realiza por medio de principios y conductas bien definidas que utilizadas adecuadamente otorgan un nuevo conocimiento o comportamiento de utilidad para la vida.

Se puede llevar a cabo el trabajo escolar de tres formas:

- Eficaz: se realiza el estudio preocupándose de la calidad de los resultados.
- Eficiente: se hace con una metodología y una técnica adecuada para alcanzar la meta.
- Efectivo: si el resultado es eficaz y eficiente y se logra el objetivo principal.
- Excelente: Si nuestro estudio está comprometido con todo tu potencial y tu forma de ser.

La excelencia en el estudio se consigue realizando un trabajo bien hecho, con metodología y técnicas adecuadas e imprimiéndole el sello personal en cuanto a valores, principios, creencias, habilidades y talentos personales.

FUENTE:

García-Huidobro C., Gutiérrez M., Condemarín E. **“A estudiar se aprende. Metodología de estudio sesión por sesión”**. Ed. Alfaomega, 4ª edición. México 2000.


ANEXO 4.1

John C. Dvorak

El fracaso del software

Desde hace poco más de una década, Microsoft se ha preocupado por la posibilidad de que, de manera súbita, su suite Office deje de venderse, lo que significaría que la compañía tendría los días contados. Sin embargo, la empresa de Redmond ha hecho todo para enfrentar tal posibilidad, pero no de la mejor manera. Aunque la paquetería mantiene sus ventas como si se tratara de pan caliente, ahora la firma intentará implementar la idea tonta de que es mejor suscribirse a un procesador de palabras en lugar de poseer uno.

En los últimos meses, la empresa de Bill Gates ha convertido en un tema importante el llamado modelo de servicio para software. Es similar a la idea que, en 1998, concibiera Scott McNealy, de Sun Microsystems: "Todo el software debería correr sobre Internet basado en un código Java". La idea provocó risas entonces.

Permítanme dejar mi posición bien marcada: la gente no querrá correr Microsoft Word, WordPad, WordPerfect o hasta WordStar sobre Internet con base en esquema de suscripción o a demanda. No es una terrible idea, pero no bajarían de manera eficiente y quizá sería muy costoso. Podría tolerar correr los programas en una red a 100 Mbps, pero ¿qué pasaría si lo hago en una red de 1 Mbps?

Microsoft tiene la idea de que debemos estar conectados a una velocidad entre los 10 y 100 Mbps. Creo que falta mucho para ello. La Red es, en general, muy lenta para que trabajen bien los esquemas de suscripción de software, y Microsoft no puede hacer nada al respecto. Para componer el problema, se ha agregado al desarrollador de Lotus Notes (el más complicado PIM/ organizador/ cliente de e-mail nunca visto) como responsable y punta de lanza de la prueba del proyecto de suscripción.

Tomen a Hotmail como modelo de Microsoft para una aplicación en línea. Cualquiera que lo use podrá imaginar la manera en que Office Online podría trabajar. Este caso de correo electrónico gratuito en línea es interesante porque cuando comenzó (en su época pre-Microsoft)

era fácil de usar, con un elegante diseño de e-mail basado en web. Pasó el tiempo y se convirtió en un horrible desorden.

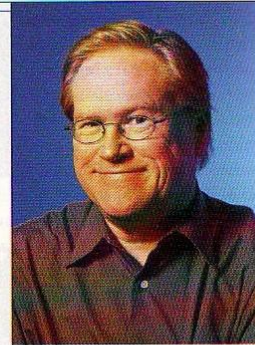
Para Microsoft, el escenario de la suscripción es la siguiente: la compañía de software más grande del mundo no puede estar celosa de compañías especializadas como SAP u Oracle, con su modo de hacer el juego del software pre-PC. Ambos casos utilizan un modelo rígido basado en pagar por las horas que uno utiliza el programa en lugar de una aplicación que venga dentro de una caja. Esta inseguridad es ilógica si se trata de la compañía que hizo que el software se empacara y fuera accesible, lo que la convirtió en la más grande y poderosa del mundo. Lamentarse del mercado que ellos crearon podría ser una tontería de magnitudes cósmicas. Es como si Boeing dejara de manera súbita la venta de jets e invirtiera en locomotoras.

Mucha de esta locura es causada por el fantasma de Google. Microsoft imagina que ellos implementarán una suite de oficina en línea antes de que Microsoft pueda defenderse y todos utilicemos 'G-Word'. Si así fuera, podríamos descargar gratis una suite de oficina de Google, pero habría que soportar anuncios al tiempo que escribimos palabras clave.

¿Por qué Microsoft no invierte el mismo esfuerzo en mejorar Office? Por ejemplo, PowerPoint es un cadáver de programa. ¡Intenten hacer algo con él! La verificación de ortografía de Word podría mejorar y disponer de opciones de traducción.

Microsoft debería levantar los viejos conceptos. Si SAP y otros paquetes similares pueden hacer dinero, la empresa de Windows debería dirigir sus energías a realizar un clon empacado de SAP. Parecería que no quieren hacer caso a los viejos modelos de obtención de ganancias, y en otros ejercicios sólo desperdiciarán dinero.

MÁS EN LA RED: Lee las columnas de John C. Dvorak cada lunes en <http://go.pcmag/dvorak> Lo puedes localizar directamente en pcmag@dvorak.org



Es como si Boeing dejara de manera súbita la venta de jets e invirtiera en locomotoras.

ANEXO 4.2

12 INTRODUCCIÓN A LAS COMPUTADORAS Y A LA PROGRAMACIÓN EN C++

CAPÍTULO 1

grammar en C++ se ha publicado 3 1/2 años después de la primera edición en inglés —coincidiendo con lo que todos esperamos sea la aceptación internacional del estándar preliminar de C++ emitido por las organizaciones ANSI/ISO.

1.10 Otros lenguajes de alto nivel

Cientos de lenguajes de alto nivel han sido desarrollados, pero sólo unos cuantos han logrado una amplia aceptación. *FORTRAN* (Traductor de Fórmulas) fue desarrollado por IBM Corporation entre 1954 y 1957 para aplicaciones científicas y de ingeniería que requieren cálculos matemáticos complejos. FORTRAN sigue siendo ampliamente utilizado, especialmente en aplicaciones de ingeniería.

COBOL (Lenguaje Común Orientado a Negocios) fue desarrollado en 1959 por varios fabricantes de computadoras y usuarios de computadoras gubernamentales e industriales. COBOL se utiliza principalmente para aplicaciones comerciales que necesitan manipular de manera precisa y eficiente grandes cantidades de información. Actualmente, más de la mitad del software de negocios sigue programándose en COBOL.

Pascal fue diseñado más o menos al mismo tiempo que C por el profesor Niklaus Wirth y se dirigió al uso académico. Hablaremos más acerca de Pascal en la siguiente sección.

1.11 Programación estructurada

Durante la década de los 60 varios grandes esfuerzos de desarrollo de software se enfrentaron a dificultades serias. Los proyectos de software generalmente se retrasaban, los costos excedían por mucho los presupuestos y los productos finales eran poco confiables. La gente comenzó a darse cuenta de que el desarrollo de software era una actividad mucho más compleja de lo que habían imaginado. Las actividades de investigación de la década de los 60 dieron como resultado el surgimiento de la *programación estructurada*, un enfoque disciplinado para escribir programas más claros que los programas no estructurados, más fáciles de probar y depurar, y más sencillos de modificar. El capítulo 2 estudia los principios de la programación estructurada. Los capítulos 3 al 5 desarrollan muchos programas estructurados.

Uno de los resultados más tangibles de esta investigación fue el desarrollo del lenguaje de programación Pascal, efectuado por Niklaus Wirth en 1971. Pascal, llamado así por el matemático y filósofo Blaise Pascal del siglo XVII, se diseñó para enseñar programación estructurada en los medios académicos y pronto se volvió el lenguaje de programación preferido en la mayoría de las universidades. Desafortunadamente, al lenguaje le faltan muchas características indispensables para que sea útil en aplicaciones comerciales, industriales y gubernamentales, por lo que no ha sido ampliamente aceptado fuera de las universidades.

El lenguaje de programación Ada fue desarrollado con el patrocinio del Departamento de Defensa de Estados Unidos durante la década de los 70 y principios de los 80. Se estaban empleando cientos de lenguajes distintos para producir los colosales sistemas de software de comando y control del Departamento de Defensa. El departamento quería un lenguaje único que cumpliera la mayoría de sus necesidades. Se seleccionó Pascal como base, pero el lenguaje Ada que resultó es bastante distinto. El nombre del lenguaje viene de Lady Ada Lovelace, hija del poeta Lord Byron. A Lady Lovelace se le atribuye haber escrito el primer programa de cómputo a principios de 1800 (para el dispositivo de cómputo mecánico llamado Máquina Analítica que diseñó Charles Babbage). Una de las capacidades importantes de Ada se llama *multitareas*; permite que los programadores especifiquen que sucederán muchas actividades al mismo tiempo. Los demás lenguajes de alto nivel



ampliamente usados que hemos descrito (incluyendo C y C++), por lo general sólo permiten que el programador escriba programas que realicen una actividad a la vez.

1.12 Fundamentos del entorno C++ típico

Los sistemas C++ por lo general consisten de varias partes: un entorno de desarrollo de programas, el lenguaje y la biblioteca estándar de C++. El siguiente análisis explica un entorno típico de desarrollo de programas C++, el cual se muestra en la figura 1.1.

Para poder ejecutar un programa C++, normalmente tiene que pasar por seis fases (vea la figura 1.1). Éstas son: *edición*, *preprocesamiento*, *compilación*, *enlace*, *carga* y *ejecución*. Aquí nos concentramos en un sistema C++ para UNIX típico (Nota: los programas de este libro se ejecutarán con pocas o ninguna modificación en la mayoría de los sistemas C++ actuales, incluidos los sistemas Windows de Microsoft). Si no está trabajando en un sistema UNIX, consulte los manuales de su sistema o pregúntele a su instructor cómo puede lograr estas tareas en su entorno.

La primera fase consiste en la edición de un archivo. Esto se logra mediante un *editor de programas*. El programador introduce un programa C++ por medio del editor y lo corrige, de ser necesario. Después el programa se guarda en un dispositivo de almacenamiento secundario, como un disco. Los nombres de los archivos de programas en C++ con frecuencia terminan con las extensiones *.cpp*, *.cxx* o *.C* (note que *C* está en mayúscula). Para obtener mayor información sobre las extensiones de los nombres de archivo, vea la documentación de su entorno C++. Dos de los editores de uso más común en UNIX son *vi* y *emacs*. Los paquetes de C++ como Borland C++ y Microsoft Visual C++ para computadoras personales tienen editores propios que se integran muy bien con el entorno de programación. Suponemos que el lector sabe cómo editar un programa.

A continuación, el programador da el comando para *compilar*. El compilador traduce el programa C++ en código de lenguaje de máquina (también conocido como *código objeto*). En los sistemas C++, antes de iniciar la fase de traducción del compilador se ejecuta automáticamente un programa *preprocesador*. El preprocesador de C++ obedece ciertos comandos especiales, llamados *directivas de preprocesador*, que indican que deben efectuarse ciertas manipulaciones al programa antes de compilarlo. Estas manipulaciones por lo general consisten en incluir otros archivos de texto en el archivo a compilar y en efectuar el reemplazo de cierto texto. En los primeros capítulos se estudian las directivas de preprocesador más comunes; en el capítulo llamado "El preprocesador" aparece el análisis detallado de todas las características del preprocesador. El compilador llama al preprocesador antes de convertir el programa a lenguaje de máquina.

La siguiente fase se llama *enlace*. Los programas C++ por lo general contienen referencias a funciones definidas en otro lado, como en las bibliotecas estándar o en las bibliotecas privadas de los programadores que trabajan en un proyecto en particular. El código objeto que el compilador de C++ genera suele contener "agujeros" debido a estas partes faltantes. El *editor de enlaces* vincula el código objeto con el código de las funciones faltantes, generando una *imagen ejecutable* (sin partes faltantes). En los sistemas UNIX típicos, el comando para compilar y enlazar un programa C++ se llama *CC*. Para compilar y enlazar un programa llamado *welcome.C*, debe escribir

```
CC welcome.C
```

en el indicador de UNIX y oprimir la tecla *Entrar* (o *Retorno*). Si el programa se compila y enlaza correctamente, el resultado es un archivo llamado *a.out*, que es la imagen ejecutable de nuestro programa *welcome.C*.

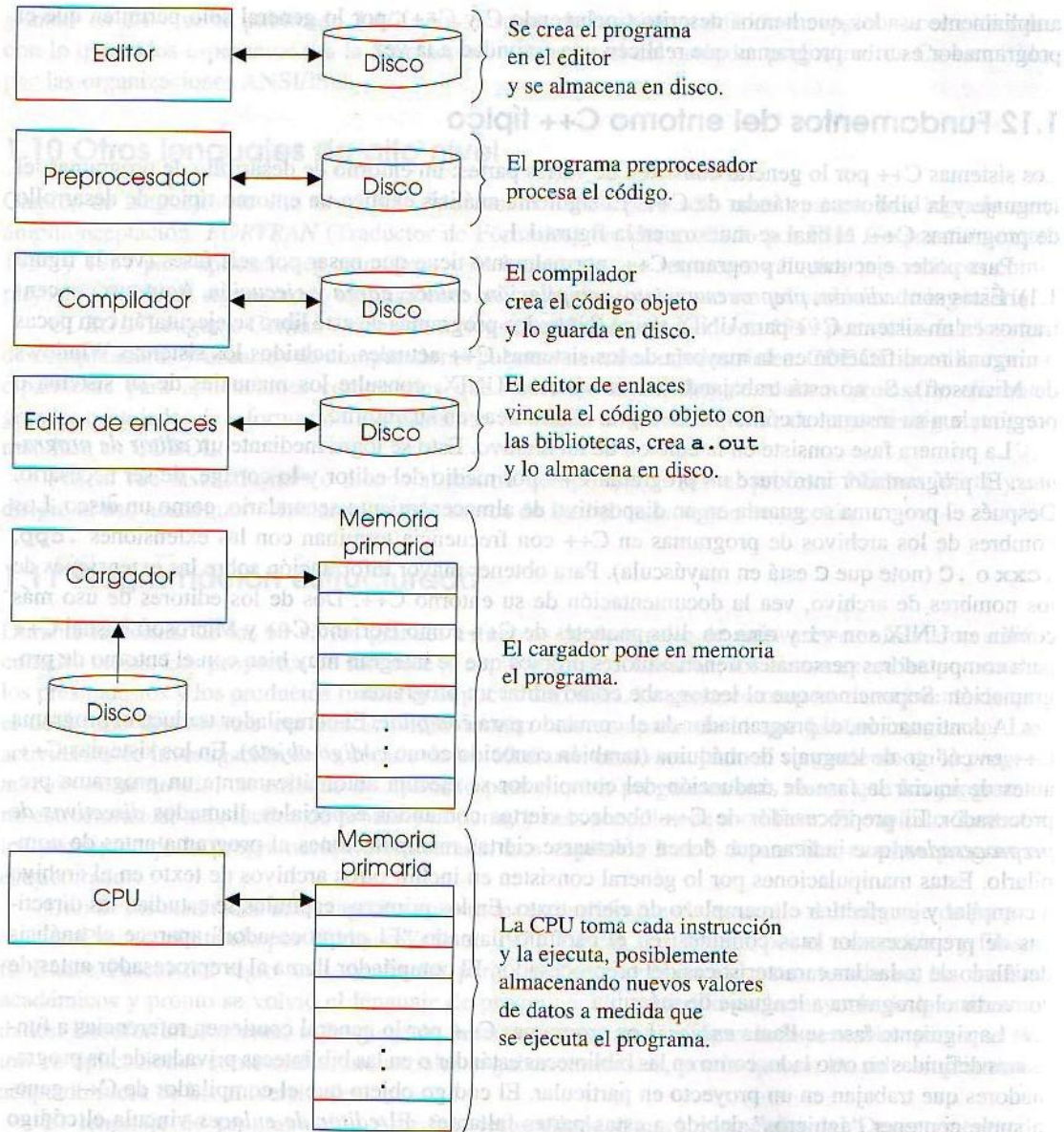


Figura 1.1 Entorno típico de C++.

Ⓢ La siguiente fase se llama *carga*. Antes de poder ejecutar un programa, debe cargarse en memoria. Esto se logra mediante el *cargador*, que toma del disco la imagen ejecutable y la transfiere a memoria.



6) Por último, la computadora, bajo el control de su CPU, ejecuta el programa, una instrucción a la vez. Para cargar y ejecutar el programa en un sistema UNIX, teclee `a.out` en el indicador de UNIX y oprima *Entrar*.

Los programas no siempre funcionan la primera vez que se prueban. Cada una de las fases previas puede fallar debido a varios tipos de errores que estudiaremos. Por ejemplo, un programa en ejecución podría intentar dividir entre cero (operación ilegal en las computadoras al igual que en la aritmética). Esto causaría que se imprimiera un mensaje de error. Entonces el programador podría regresar a la fase de edición, efectuar las correcciones necesarias y repetir las fases para determinar que las correcciones funcionan de manera apropiada.



Error común de programación 1.1

Los errores como la división entre cero suceden durante la ejecución del programa, por lo que se llaman errores en tiempo de ejecución. La división entre cero por lo general es un error fatal, es decir, un error que provoca que el programa termine de inmediato sin haber concluido de manera apropiada su trabajo. Los errores no fatales permiten que los programas se ejecuten hasta el fin, generalmente produciendo resultados incorrectos. (Nota: en algunos sistemas, la división entre cero no es un error fatal. Lea la documentación del sistema con el que esté trabajando.)

Muchos programas en C++ efectúan entradas y salidas de datos. Ciertas funciones de C++ toman su entrada de `cin` (el flujo de entrada estándar), que normalmente es el teclado, pero el `cin` puede conectarse a otro dispositivo. Es común que la salida sea hacia `cout` (el flujo de salida estándar), que por lo general es la pantalla de la computadora, pero `cout` puede conectarse a otro dispositivo. Cuando decimos que un programa imprime un resultado, lo que normalmente queremos decir es que el resultado se presenta en pantalla; la salida de los datos puede ser hacia otros dispositivos de salida, por ejemplo a disco o impresora. También hay un flujo de errores estándar conocido como `cerr`. El flujo de `cerr` (normalmente conectado a la pantalla) sirve para desplegar mensajes de error. Es común que los usuarios dirijan la información de salida normal, es decir `cout`, a un dispositivo distinto de la pantalla mientras mantienen asignado `cerr` a la pantalla, de modo que puedan enterarse de inmediato que han sucedido errores.

1.13 Apuntes generales sobre C++ y esta obra

C++ es un lenguaje complejo. Los programadores experimentados de C++ a veces se enorgullecen de su capacidad para utilizarlo de manera extraña, torcida y embrollada. Esto hace que los programas sean más difíciles de probar y depurar, y más complejos de adaptar a requisitos cambiantes. Este libro se dirige a los programadores novatos, por lo que subrayamos la *claridad*. La siguiente es nuestra primera “buena práctica de programación”.



Buena práctica de programación 1.1

Escriba sus programas de C++ sencilla y directamente. Esto a veces se conoce como “mantenlo simple”. No “estire” el lenguaje intentando usos raros.

Incluiremos muchos de estos tips a lo largo de la obra para subrayar las prácticas que ayudarán a escribir programas más claros, comprensibles y fáciles de mantener, probar y depurar. Estas prácticas son sólo guías; sin duda usted desarrollará su propio estilo de programación. También subrayaremos varios errores comunes de programación (problemas que prever para evitar caer en estos errores en sus programas), tips de desempeño (técnicas que le ayudarán a escribir programas más rápidos y de



ANEXO 4.3

Karl Jentjens

¿Mac, intel, o ambos?

En junio de 2005, Steve Jobs hizo el anuncio oficial: Apple haría a un lado al procesador PowerPC para dar lugar a una nueva generación de sistemas basados en plataforma Intel. Para los usuarios de la marca, la noticia cayó como balde de agua fría; de inmediato, surgieron muchas dudas en relación con la arquitectura, estabilidad, compatibilidad, seguridad y facilidad de uso de las nuevas Mac-Intel.

La decisión de cambiar la plataforma de las máquinas de Apple, más que comercial, fue una cuestión tecnológica. Según el propio Jobs, se dio porque "Apple tiene la convicción de fabricar las mejores computadoras", algo que no se ha cumplido en últimas fechas. La compañía prometió, en 2003, la disponibilidad de un procesador a 3GHz para mediados de 2004 y -en 2006- aún no lo tiene. Tampoco ha podido lanzar la tan esperada PowerBook G5.

Los ejecutivos de Apple no están satisfechos con el desempeño que ha mostrado IBM en el desarrollo de nuevos procesadores y están convencidos de que Intel puede lograr un progreso más rápido para los sistemas Mac a futuro.

Mientras que en las 'altas esferas' se habla de negocios y tecnologías, los usuarios especulan acerca de todo lo que este cambio de plataforma podría implicar: ¿su software actual es compatible con las nueva Mac-Intel? ¿Será conveniente actualizar la máquina de inmediato? ¿Será una transición sencilla o causará problemas?

Además, celosos de su plataforma, preguntan si, ahora, cualquier Mac podrá ejecutar Windows o -peor aún para ellos- si tendrán que compartir el OS X con los usuarios de PC.

La respuesta a estas preguntas ya está sobre la mesa. La mayoría de las aplicaciones pueden ejecutarse en los sistemas basados en Intel; aquellas que son 'Universales', incluso trabajan en promedio 1.2 a 1.3 veces más rápido que sobre plataforma G5. El problema se da con aplicaciones desarrolladas para el procesador PowerPC, que se valen de Rosetta (que no es otra cosa que un emulador, aunque Apple ha tenido mucho

cuidado en no mencionar este término), que trabajan casi a la mitad de la velocidad que lo hacen en un sistema G5.

Si hablamos de instalar Windows en una Mac, a principio de cuentas... ¿a quién podría interesarle? Siempre ha sido más divertido jugar con las entrañas de una PC que con las de una Mac, por la variedad de componentes disponibles y la flexibilidad de escalar a nivel interno y externo la máquina. Por tanto, para el caso, me quedo con Windows en una PC.

¿Mac OS en una PC? Tal vez, por un rato... Pero al ser usuario de la PC casi desde sus inicios (en un par de meses viene su veinticinco aniversario), me ha costado mucho trabajo adaptarme este último año a usar una Mac en la oficina; tanto, que a la fecha tengo una PC instalada junto a ésta para realizar la mayoría de mis actividades, excepto las que, por fuerza, requieren que utilice la Mac.

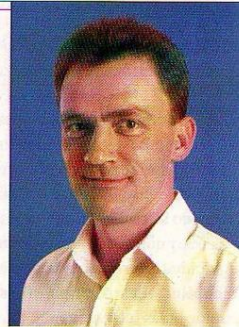
Es cierto, Mac OS X ofrece una interfaz muy agradable y su facilidad de uso es impresionante, pero borrar el tatuaje de PC que llevo en las venas, ha sido imposible. La facilidad que ofrece la Mac es tanta, que resulta molesta. Prefiero todavía las complejidades de la PC.

Una cosa sí es cierta. A pesar de que el móvil del rediseño de las Mac es el esfuerzo tecnológico que hace Apple por ofrecer mejores sistemas a sus usuarios, seguro se verá un incremento importante en las utilidades de la compañía, lo cual es bueno para todos.

La mejor calidad de un producto se logra a través de un buen sistema de investigación y desarrollo, mientras que este último sólo puede darse cuando existen finanzas sanas.

Tomando en cuenta las ventajas tecnológicas y económicas de esta nueva alianza entre Apple e Intel -como dice su campaña-, sólo imagina las posibilidades de esta unión.

MÁS EN LA RED: puedes contactar a Karl-Heinz Jentjens Kraus en karlheinz@editorial.televisa.com.mx Para más información sobre sus columnas, visita www.pcmagazine.com.mx



Mac ofrece una interfaz agradable y su facilidad es impresionante, pero borrar el tatuaje de PC que llevo en las venas, ha sido imposible.



ANEXO 5

CANAL PANDA



Pharming, nueva técnica de fraude



FERNANDO DE LA CUADRA

Los hackers están intentando, cada vez más, obtener beneficios económicos de sus actuaciones y del malware que crean.

Si hasta ahora uno de los fraudes más extendidos era el phishing, consistente en engañar a los usuarios para que efectúen operaciones bancarias en servidores web con el mismo diseño que un banco on-line, el pharming entraña aún mayores peligros que el phishing.

Básicamente, el pharming consiste en la manipulación de la resolución de nombres en Internet, llevada a cabo por algún código malicioso que se ha introducido en el equipo. Cuando un usuario teclea una dirección (como puede ser www.pandasoftware.com), ésta debe ser convertida a una dirección IP numérica, como 62.14.63.187. Esto es lo que se llama resolución de nombres, y de ello se encargan los servidores DNS, siglas que corresponden a "Domain Name Server". En ellos se almacenan tablas con las direcciones IP de cada nombre de dominio. A una escala menor, en cada ordenador conectado a Internet hay un fichero en el que se almacena una pequeña tabla con nombres de servidores y

direcciones IP, de manera que no haga falta acceder a los DNS para determinados nombres de servidor, o incluso para evitarlo.

El pharming consiste en modificar este sistema de resolución de nombres, de manera que cuando el usuario crea que está accediendo a su banco en Internet, realmente está accediendo a la IP de una página web falsa.

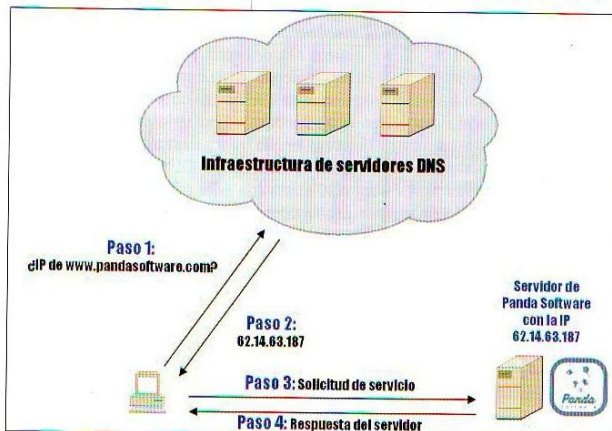
El phishing debe su éxito a la ingeniería social, aunque no todos los usuarios caen en estos trucos y su éxito está limitado. Y además, cada intento de phishing se debe dirigir a un único tipo de servicio bancario, por lo que las posibilidades de éxito son muy limitadas. Por el contrario, el pharming puede atacar a un número de usuarios de banca muchísimo mayor.

Además, el pharming no se lleva a cabo en un momento concreto, como lo hace el phishing mediante sus envíos, ya que la modificación de DNS queda en un ordenador, a la espera de que el usuario acceda a su servicio bancario. De esta manera, el atacante no debe estar pendiente de un ataque puntual, como hemos mencionado antes.

La solución para esta nueva técnica de fraude pasa, de nuevo, por las soluciones de seguridad antivirus. Las acciones necesarias para llevar a cabo el pharming necesitan efectuarse por alguna aplicación en el sistema a atacar (puede ser un fichero exe, un script, etc.). Pero antes de poder ejecutarse esta aplicación, debe llegar al sistema objetivo, evidentemente. La entrada de código en el sistema puede ser a través de múltiples vías, tantas como entradas de información hay en un sistema: el e-mail (la más frecuente), descargas por Internet, copias desde un disco o CD, etc. En todas y cada una de estas entradas de información, el antivirus debe detectar el fichero con el código malicioso y eliminarlo, siempre que se encuentre detectado como una aplicación dañina dentro del fichero de firmas de virus del antivirus.

Desgraciadamente, hoy en día nos movemos en un escenario en el que el malware ha adquirido una velocidad de propagación muy elevada, y los creadores son más y ofrecen al resto de la comunidad hacker los códigos fuente para que introduzcan variaciones y puedan crear ataques nuevos. Los

Esquema del proceso de resolución de nombres cuando accedemos a los servicios del servidor de Panda Software.





laboratorios de virus no tienen tiempo suficiente para efectuar la detección y eliminación del malware para todos los nuevos códigos antes de que lleguen a propagarse en unos pocos PCs. A pesar de los esfuerzos y la mejora de los laboratorios, es humanamente imposible que se elabore una solución adecuada y a tiempo para algunos códigos que se propagan en cuestión de minutos.

La solución para este tipo de amenazas no debe ser, al menos en un primer frente de protección, una solución reactiva, sino que deben instalarse sistemas mediante los cuales se detecten no los ficheros en función de firmas viricas, sino las acciones que se llevan a cabo en el ordenador. De esta manera, cada vez que se intente realizar un ataque al sistema de DNS del ordenador (como es el caso de las aplicaciones para pharming), sea reconocido el ataque y detenido, así como el programa que lo ha llevado a cabo, bloqueado.

Sin embargo, existe un peligro añadido a esta nueva técnica de pharming, que reside en los servidores proxies anónimos. Muchos usuarios desean ocultar su identidad (su dirección IP) a la hora de navegar, por lo que utilizan servidores proxy instalados en Internet que llevan a cabo la conexión con la IP del servidor en lugar de la IP del cliente. En el peor de los casos, uno de estos servidores proxy puede tener la resolución de nombres alterada, de manera que los usuarios que intenten entrar en su página bancaria (a pesar de que su sistema local está perfectamente asegurado) sean redirigidos por el proxy a una página con el mismo diseño y apariencia de su banco, pero falsa. También podríamos pensar, siendo más positivos, que el servidor proxy ha sufrido algún tipo de ataque que altere su sistema de resolución de nombres de dominio.

En cualquiera de los casos, el problema del pharming se plantea como peligroso, aunque de muy fácil solución. Únicamente con sistemas capaces de detectar los cambios en la resolución de nombres de Internet en el ordenador y con sistemas para su bloqueo, podremos hacer frente a la avalancha de códigos maliciosos que nos espera y que intentan estafar a los usuarios. ☺

Sobre el autor

Fernando de la Cuadra (fdelacuadra@pandasoftware.com) es editor técnico internacional de Panda Software (<http://www.pandasoftware.com>).

Los usuarios de banca on-line amenazados por un nuevo ladrón de contraseñas



El laboratorio anti-virus de Panda Software (PandaLabs) ha detectado recientemente la aparición de Searchmeup, el primer adware (o programa que muestra publicidad) que utiliza una vulnerabilidad de Microsoft para sustraer información de los usuarios.

Los usuarios de banca on-line se ven expuestos a esta amenaza simplemente al navegar por Internet. Este adware se introduce en las máquinas sin el consentimiento del usuario utilizando una vulnerabilidad denominada Exploit/LoadImage.

Una vez introducido en el ordenador cambia la página de inicio del navegador, dejando en su lugar la de un buscador que, cada vez que se carga, muestra popups que instalan spyware y dialers en el equipo como el troyano Tofger.AT, los dialers Dialer.BB y Dialer.NO, y otro adware denominado Adware/TopConvert.

Infectado el ordenador, cuando el usuario abre Internet Explorer, Tofger.AT se activa y si el usuario visita uno de los 15 bancos on-line más importantes de España (páginas con conexiones seguras HTTPS) Tofger.AT capturará la contraseña utilizada para acceder a las siguientes entidades bancarias: cajamadrid, bpinet, millenniumbcp, hsb, barclays, lloydtsb, halifax, autorize, bankofamerica, bancodevalencia, cajamar, portal.com, bancaja, caixagalicia, caixapenedes, ebankinter, caixabadell, bes, banif, millenniumbcp, totta, banco-mais, montepiogeral, bpinet, patagon, lacaixa, citibank, bbvanet, banesto, e-trade y unicaja. El usuario infectado no es consciente de la existencia de Searchmeup a menos que:

- Observe el cambio de la página de inicio de su navegador.
- Si se genera un error y se reinicia el ordenador en un minuto (lo cual puede no ocurrir siempre).
- También puede mostrar pantallas de error de sistema (pantallazos azules).

La vulnerabilidad Exploit/LoadImage de la que se aprovecha Searchmeup afecta a ordenadores con sistema operativo Windows 2003/XP/



2000/NT/Me/98, y permite ejecutar código arbitrario en el sistema. Puede ser aprovechada por un atacante alojando un cursor o icono especialmente creado en una página web maliciosa, o en un mensaje de correo electrónico en formato HTML. Para corregir este problema Microsoft ha publicado la correspondiente actualización, cuya instalación se recomienda. Para más información, puede visitarse la dirección <http://www.microsoft.com/spain/technet/seguridad/boletines/MS05-002-IT.msp>.

Ante el peligro que representa Searchmeup y el troyano Tofger.AT, Panda Software recomienda extremar las precauciones y mantener actualizado el software antivirus. Los clientes de Panda Software ya tienen a su disposición las correspondientes actualizaciones para la detección y desinfección de estos nuevos códigos maliciosos.

Asimismo, los clientes de Panda Software ya tienen disponibles las actualizaciones para instalar las nuevas Tecnologías TruPrevent junto con su antivirus y estar, así, protegidos de forma preventiva frente a nuevos códigos maliciosos. Por otro lado, para usuarios que cuenten con otros antivirus del mercado, Panda TruPrevent Personal es la solución idónea, ya que es compatible y complementaria a éstos y proporciona una segunda línea de defensa y una protección preventiva que actúa mientras el antivirus es actualizado, disminuyendo el riesgo de ser infectados. El lector puede conocer más sobre las tecnologías TruPrevent en <http://www.pandasoftware.es/truprevent>.

Para la detección y desinfección gratuita de los ordenadores, los usuarios pueden utilizar el antivirus on-line Panda ActiveScan, disponible en <http://www.pandasoftware.es>.

ANEXO 6

Instrucciones: Rellena con tu color favorito los recuadros que contienen información verdadera sobre las técnicas adecuadas para preparar exámenes.

Considerar el % de la calificación que representa mi examen.

Es de gran ayuda ordenar los temas y subtemas que vendrán en mi examen.

Es imprescindible sobrevalorar el examen, para –de esa forma– presionarme y rendir mejor.

¡Se deben leer velozmente todas las preguntas del examen!

Es aconsejable prestar atención a mi escritura, pues mi letra cambia la percepción de mi maestro sobre mí.

Es recomendable predisponerme al fracaso para, así, atemorizarme y darle mucho valor al examen y obtener buenas notas.

La atención resulta básica, pues en ocasiones por las prisas, pasamos preguntas y ni siquiera nos damos cuenta de que nos saltamos algunos ítems.

Si no entiendo una pregunta, lo mejor es repararla y repararla antes de pasar a otra hasta que la entienda.

Es importante tomar en cuenta que los maestros son muy estrictos y desean reprobarnos.

Es recomendable realizar mucho ejercicio físico y cansarme un día antes del examen.

Para los exámenes prácticos es básico contar con todo el material necesario antes de comenzar su aplicación.

Es mejor presentar un examen en ayunas.

Si no recuerdo un dato, es adecuado presionar mucho a mi mente para que se acuerde de dicha información, y esforzándola así lograré recordar.

Una noche antes del examen, es importante dormir más horas de las que acostumbro descansar.

El repaso es una buena técnica para mejorar mi aprendizaje previo al examen.

Una actitud de confianza y convicción es un gran aliado para presentar mis exámenes.

Un día antes del examen, debo pasarme estudiando todo el tiempo posible.

No importa el lugar donde estudie para el examen, lo importante es estudiar.