Clase 4

Sitio:	Instituto Superior de Estudios Pedagógicos
Curso:	Taller de Administración y Configuración de Herramientas Tecnológicas - PROGRAMACION-HERRAMIENTAS-Ago17-01-IESSB
Libro:	Clase 4
Imprimido por:	Javier Agustín Ferreira
Día:	miércoles, 20 de septiembre de 2017, 21:04

Tabla de contenidos

- INTRODUCCIÓN
- DESARROLLO: Arqueología computacional
- Contra la obsolescencia tecnológica programada
- Contra las limitaciones al hardware impuestas por el software
- A modo de CIERRE
- ACTIVIDADES
- MATERIALES

Taller de Administración y Configuración de Herramientas Tecnológicas

Especialización Docente de Nivel Superior en Enseñanza de la Programación en Informática

CLASE 4 LIBERAR COMPUTADORAS: EXPERIENCIAS Y PRÁCTICAS DE APROPIACIÓN TECNOLÓGICA

INTRODUCCIÓN

A través de las clases y las prácticas presenciales comprendimos que nos podemos apropiar tanto del hardware como del software y aumentar el grado de libertad para interactuar con la tecnología.

Hoy las computadoras toman muchas formas y son ubicuas. Esto ya se vio tanto en este módulo como en el anterior: una placa Arduino también es una computadora.

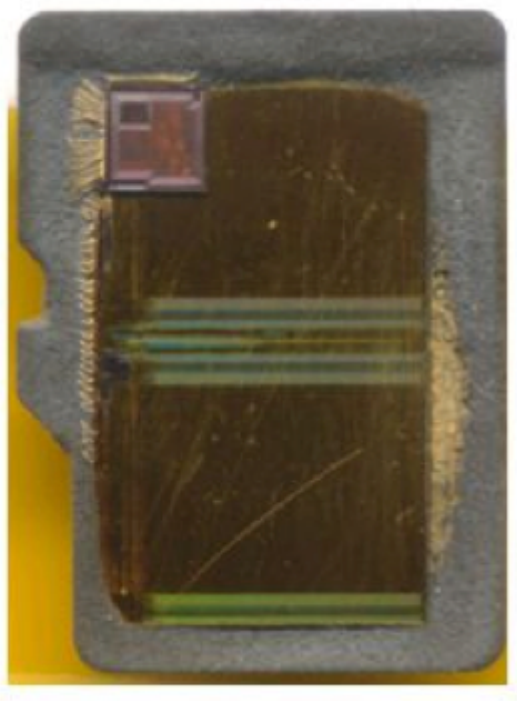
En esta última clase recolectaremos experiencias y las expondremos para mostrar que el fenómeno de apropiación va más allá de lo técnico y tiene implicancias para el ejercicio de nuestra libertad tecnológica.

Taller de Administración y Configuración de Herramientas Tecnológicas

Especialización Docente de Nivel Superior en Enseñanza de la Programación en Informática

Arqueología computacional

Como hemos aprendido hasta aquí, el concepto de computadora tiene muchas formas, y va desde una tarjeta SD hasta una supercomputadora.



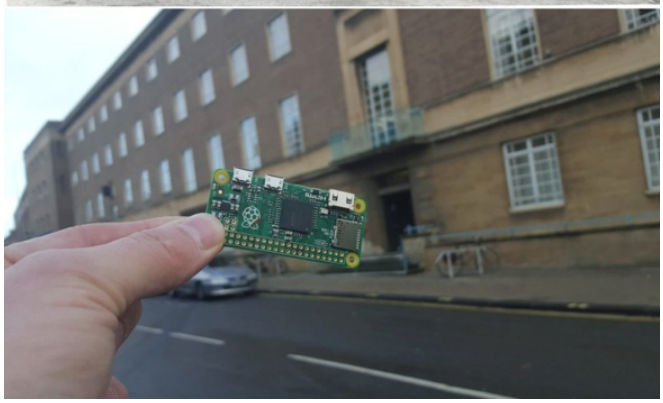
Tarjeta de memoria **microSD** al desnudo. En la parte superior izquierda está la CPU y la RAM y todo el resto es la memoria flash

Supercomputadora Mendieta del CCAD-UNC.

Aunque lo que muestran las imágenes son los extremos, vimos varios ejemplares al medio: el Arduino de los robots, el punto de acceso Wi-Fi, la netbook y la PC de escritorio. **Todas son computadoras y por lo tanto desarmables, modificables, reparables en hardware y totalmente programables en software.**

Las computadoras no solo son invariantes respecto de su tamaño, también lo son respecto de la época, no importa si es una computadora de 1978 o de 2017, la funcionalidad básica es la misma. Esta es una forma de obsolescencia programada y se denomina **obsolescencia percibida**.

No habría nada de malo en montar un laboratorio de **Talent DCP-200** MSX para enseñar a programar a nuestros estudiantes, aunque los equipos tengan más de 30 años, de hecho, una **BBC micro:bit**, que actualmente se distribuye en todos los estudiantes de entre 11 y 12 años del Reino Unido, tiene aún menos prestaciones que esta computadora que se compra en ferias de pulgas.



En la primera imagen vemos una Elliot 405 de 1957.

En la segunda, una Raspberry Pi Zero de 2015, ambas fotografías tomadas frente a la Municipalidad de Norwich, Inglaterra.

La integración y la potencia de cálculo aumentaron millones de veces.

Gustavo del Dago (2012), arqueólogo computacional de Argentina, define la **arqueología computacional** como conjunto de tareas y actividades relacionadas con la preservación, el estudio y análisis de los sistemas de computación antiguos, aportando nuevas fuentes documentales para la historia de la computación.

Es interesante conocer que en Córdoba se diseñó, fabricó y comercializó la primera microcomputadora argentina, la Microsistemas MS101. A continuación, les acercamos dos artículos y un video sobre los cuales les proponemos algunas preguntas para tomar nota:

- **Microsistemas, ese hito olvidado de la computación argentina**.

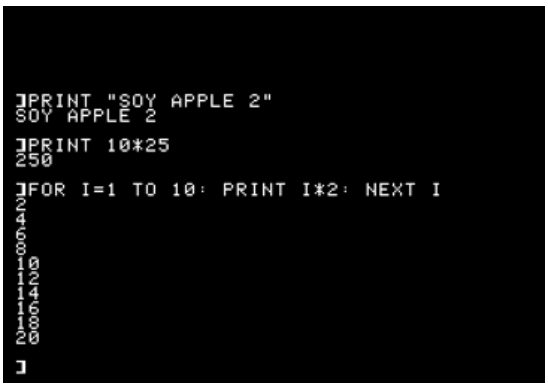
- **Microsistemas y la pionera MS101, el sueño del Silicon Valley cordobés**.

• ¿Qué capacidades de memoria y disco tenía esta primera microcomputadora?
• ¿Cuál era su función?
• ¿Era programable?
• ¿Tenía almacenamiento permanente?
• ¿Cuál era su precio? ¿Y a valores actuales?
• ¿Cómo se compara con la Apple][que es casi del mismo año?

De la misma manera que vimos que con VirtualBox una PC puede emular una PC a la perfección, existen muchos programas emuladores de computadoras antiguas para mantener vivo el ecosistema de la máquina y sus programas. La emulación resulta casi exacta gracias al gran poder de cómputo de las máquinas actuales, y esto permite lograr emular hasta el efecto del haz de electrones sobre fósforo del tubo de rayos catódicos. Podemos entonces realizar un viaje de arqueología computacional hasta 1978 y emular con **apple][js** de manera perfecta una **Apple][**, la computadora que inició el imperio situado en Cupertino, California, EEUU.

A continuación les proponemos indagar acerca de las características técnicas de la computadora: cantidad de memoria RAM, capacidades de video y sonido, sistema de almacenamiento permanente, precio de venta histórico y valores actuales. ¿Qué sucede cuando la máquina enciende? ¿Que "sistema operativo" carga? Cuidado, para poder operar hay que apretar la tecla `RESET`.

• Estudiar un poco de **lenguaje BASIC**. Utilizar el lenguaje BASIC como una calculadora y lenguaje de programación. ¡En algunas cosas es como Python!



• El emulador viene con una pequeña biblioteca de programas. A través del emulador de la **disquetera**, insertamos en el *Disk 1*, una imagen de un programa y reiniciamos la máquina con el botón de *power*. ¿Cómo eran los juegos de esa época? ¿Y el software educativo?

Por ejemplo: utilizando el **emulador online de Apple][** que corre dentro de un navegador, copiamos el programa *Rod's Color Pattern* de la página 55 del **manual de usuario de la computadora** y al correr con el comando `RUN` el programa de la izquierda, sale la figura de la derecha.

```
100 GR
105 FOR W=3 TO 50
110 FOR I=1 TO 19
115 FOR J=0 TO 19
120 K=I+J
130 COLOR=J*3/(I+3)+I*W/12
135 PLOT I,K: PLOT K,I: PLOT 40
   -I,40-K
136 PLOT 40-K,40-I: PLOT K,40-I:
    PLOT 40-I,K: PLOT I,40-K: PLOT
   40-K,I
140 NEXT J,I
145 NEXT W: GOTO 105
```

Aunque la práctica de **copiar programas de forma manual** sea una rareza en este mundo de Internet, algunos investigadores en tecnologías resaltan sus virtudes, ya que mejora el proceso de lecto-comprensión del código y ayuda a entender el lenguaje de programación (Montfort, 2016).
El manual de Apple][tiene muchos programas para copiar, y esto puede ser una actividad interesante para llevar a los estudiantes.

¿Cómo se llega de una computadora de 39 años que nos permite aprender los rudimentos de la programación a una computadora que es millones de veces más poderosa, pero que cada dos años “no sirve más”, “se ve demodé” y además no se puede programar? La respuesta está en la obsolescencia programada y la obsolescencia percibida.

Contra la obsolescencia tecnológica programada

Hay un ejemplo paradigmático donde todos los conceptos aprendidos en este módulo se condensan en un solo aparato: el smartphone. Estas computadoras de bolsillo tienen una potencia de cálculo, memoria RAM y almacenamiento permanente comparables con cualquier computadora portátil. Un celular de 4 núcleos a 2 GHz, 2 GiB de RAM y 16 GiB de memoria flash es un equipo de **rango medio** como nos dicen en los innumerables casas de celulares donde nos venden estas tecnologías de comunicación y computación atadas a un plan de telefonía.

En una **aparición televisiva de 2015**, el personaje Emmet “Doc” Brown (Christopher Lloyd) de la trilogía “Volver al Futuro” descubre que el mayor avance desde 1985 a la fecha no son los autos voladores, sino las **pequeñas supercomputadoras portátiles para telecomunicaciones** que “le permitirían a los físicos triangular complejas ecuaciones en tiempo real”. Sin embargo, el anfitrión del show le cuenta que solo se utilizan para sacarse selfies y enviar emojies.

Tal vez la situación sea más triste aún. Estas pequeñas supercomputadoras portátiles para telecomunicaciones sufren de:

1) Un *lock-in* a la empresa telefónica a la cual pertenece, a menos que sea “liberado”.
2) La imposibilidad de cambiar su sistema operativo, a menos que se obtenga el código de desbloqueo (*unlock code*).
3) La obsolescencia programada del sistema operativo que les da funcionalidad.

Es muy común encontrar en nuestros hogares teléfonos celulares con buena cantidad de núcleos, memoria RAM y almacenamiento permanente que “ya se quedaron viejos” porque el sistema operativo no se actualizó y no es posible utilizar apps modernas. O peor aún, que nuestros hijos descartan porque no lucen modernos como los de sus amigos y amigas. ¿Qué está pasando? Es simple, no nos estamos apropiando del aparato, parecería que el celular lo obtuvimos en comodato compartido entre el fabricante y la empresa de telefonía.

Existe la posibilidad de apropiarnos de los celulares, que en definitiva son computadoras y que, en su mayoría, corren Android que es un GNU/Linux con una interfaz gráfica diseñada para pantallas sensibles al tacto. Gracias a que **Android es un proyecto de software libre**, es posible recompilar cualquier versión de Android para nuestro celular y cambiar a la última disponible para el equipo cuyo fabricante ya dictaminó la obsolescencia.
Hay muchas alternativas y se denominan ***custom ROMs***:

- **Paranoid Android**
- **OmniROM**
- **Carbon ROM**

No solamente estas *custom ROMs* nos permiten tener una versión de Android más moderna y por lo tanto con menos problemas de seguridad y mayor funcionalidad, sino que además suelen estar desprovistas de todas esas aplicaciones que nos obligan a utilizar los fabricantes de los equipos y las compañías telefónicas que los adquieren y nos los re-venden. Por lo tanto, cambiar a una ROM, no solo hace que nuestra pequeña supercomputadora portátil para telecomunicaciones se vea nueva con el último software, sino que además suele funcionar más rápido.

Contra las limitaciones al hardware impuestas por el software

Apropiarse del hardware y software no solo nos previene de la obsolescencia tecnológica programada, sino que nos permite obtener más prestaciones del mismo hardware. Tal vez el ejemplo paradigmático sea un viejo chiste:

- ¿Qué antivirus me recomendás para Windows?
- Linux

Y aunque ya vimos que GNU/Linux no es invulnerable, permanecer en el 5% del mercado nos mantiene protegidos de males mayores. O para ponerlo en otros términos, utilizar el sistema operativo más popular implica que tenemos que usar programas denominados antivirus que nos quitan CPU, RAM y disco duro.
Hay ejemplos aún peores.
Tomemos el punto de acceso marca TP-Link modelo **TL-WA701ND**. Este aparato es un dispositivo económico que nos permite a partir de una entrada de red cableada denominada Ethernet, compartirla a través del éter por Wi-Fi en todo nuestro hogar. A lo mejor organizamos una reunión y notamos que aunque le ofrecemos nuestra conexión Wi-Fi a nuestros invitados, ellos no la pueden usar, mientras que en el día a día el aparato funciona de mil maravillas. Probablemente terminemos diciendo que es un aparato pequeño para uso hogareño y que no se aguanta tantas conexiones, y apenas tengamos oportunidad compraremos alguno más caro con dos o tres antenas.
La limitación es del software y no del hardware como indica el artículo "**WLAN-Basis nimmt nur 15 Nutzer an**" (**traducción automática al español**) de la revista alemana Magazin für Computertechnik (**c't**).
¡La solución es apropiarnos del aparato y cambiar el software que lo controla y limita artificalmente! La distribución de GNU/Linux llamada **OpenWRT** está diseñada para puntos de acceso como este y permite "flashear su ROM", es decir, **cambiar el sistema operativo por uno alternativo**.
No solamente la limitación a 15 conexiones desaparece, sino que además ahora podemos elegir la potencia de irradiación de la antena y aumentar el cubrimiento Wi-Fi con **exactamente el mismo aparato**.

A modo de cierre

Más que un cierre hacemos una apertura: las técnicas, las ideas y los ideales ya están descritos, de ahora en más el hardware y el software son nuestros.

Para descargar los contenidos de esta clase en formato PDF **haga clic aquí**.

Referencias

• Apple Computer Inc. Apple][Reference Manual 1978. Disponible en: **https://ia600504.us.archive.org/12/items/applerefjan78/applellrefjan78.pdf**

• De la Vega, C. (2107). Microsistemas, ese hito olvidado de la computación argentina. En TSS. Universidad Nacional de San Martín. Disponible en: **http://www.unsam.edu.ar/tss/**

• Del Dago,G. (2012). Creating an software ecosystem in order to preserve the first Argentine computer language and compiler: A case study on computing archaeology. En *Informática (CLEI)*, 2012 XXXVIII Conferencia Latinoamericana. Disponible en: **http://dblp.uni-trier.de/pers/hd/d/Dago:Gustavo_del**

• Magazin für Computertechnik (2013). WLAN-Basis nimmt nur 15 Nutzer an. Disponible en: **https://www.heise.de/ct/hotline/WLAN-Basis-nimmt-nur-15-Nutzer-an-2056906.html** (traducción automática al español)

• Montfort, N. (2016). Exploratory Programming for the Arts and Humanities. The MIT Press. Disponible en: **http://ieeexplore.ieee.org/document/6427261/?reload=true**

Cómo citar este material:

Wolovick, N. y Equipo de producción de materiales educativos en línea (2017). Clase 4: Liberar computadoras: experiencias y prácticas de apropiación tecnológica. Módulo *Taller de administración y configuración de herramientas tecnológicas*. Especialización Docente de Nivel Superior en Enseñanza de la Programación en Informática. Córdoba: Instituto Superior de Estudios Pedagógicos - Ministerio de Educación de la Provincia de Córdoba.

Esta obra está bajo una licencia Creative Commons (CC BY-NC-SA 3.)

Actividad de acreditación 4: Ejemplos de apropiación del hardware y/o software

Objetivo

- Profundizar en los planteos y conceptos presentados en el módulo a partir de recabar experiencias de apropiación de hardware y software.

Consignas

Recupere una experiencia escolar, hogareña o empresarial, en lo posible cercana y significativa, de apropiación de hardware y/o software. Nárrela en el fotolog grupal en una entrada que no supere las 400 palabras. El caso presentado debe tener algún tipo de relación con las experiencias y los planteos conceptuales del módulo.

Criterios de evaluación

Se valorará especialmente:

- Que la historia sea compatible con las experiencias vividas durante el módulo.
- Que la historia resulte significativa para algún actor de la comunidad educativa.
- Que no solo se muestre lo positivo, sino también las dificultades y soluciones.

Plazo de trabajo: indicado en la **Hoja de ruta**.

Haga clic acá para encontrar los Materiales de la clase 4.