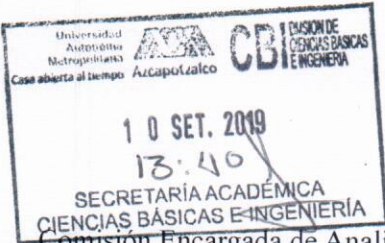


No. 6
Ser. 10. 19



México, Ciudad de México, a 10 de septiembre de 2019

Comisión Encargada de Analizar las Propuestas de Adecuación, Modificación y Creación de los Planes y Programas de Estudio de Licenciatura de la División de Ciencias Básicas e Ingeniería-Azcapotzalco

PRESENTE

En el marco de la consulta a la comunidad de Ciencias Básicas e Ingeniería, Unidad Azcapotzalco en relación a la propuesta de Adecuación al Plan y Programas de Estudio del Tronco General de las Licenciaturas de la DCBI-A, escribo este documento para expresar mi opinión respecto a la propuesta de adecuación al programa de estudios de la UEA "Programación Estructurada".

Los cambios que propone el Grupo Temático de "Cómputo General" a la UEA "Programación Estructurada" y que son a su vez los que publicó la División para su discusión, son los siguientes:

1. Eliminar la seriación con la UEA "Introducción al Cálculo".

La justificación dada por el Grupo Temático es que se desea flexibilizar los prerrequisitos de "Programación Estructurada". El problema es que la flexibilización citada no es visible para un alumno regular (las dos UEA prerrequisito "Introducción al Cálculo" y "Complementos de Matemáticas" suelen cursarse en el mismo trimestre, ya que ambas dependen sólo de "Taller de Matemáticas" del Tronco de Nivelación) además de que la cantidad de alumnos no regulares con posibilidades de inscribir "Programación Estructurada" aumentaría dramáticamente; según el documento *Datos sobre la evolución y estado actual de las licenciaturas de la División de CBI-A* anexo a la propuesta de adecuación del Tronco General, la UEA "Introducción al Cálculo" es la que más NA acumuladas tiene. Actualmente la demanda de grupos de "Programación Estructurada" es muy alta y al Departamento de Sistemas no siempre le ha sido fácil cubrirla (la impartición correcta de esta UEA requiere el uso de salones con computadoras para los alumnos). Eliminar la seriación con "Introducción al Cálculo" podría exacerbar seriamente el problema de oferta-demanda para "Programación Estructurada". Por otra parte, el concepto de función se cubre en "Introducción al Cálculo" y es útil que el alumno lo tenga presente al cursar "Programación Estructurada".

2. Incorporar Python a la lista de lenguajes de programación a enseñar.

La justificación dada por el Grupo Temático es que el uso de Python ha aumentado considerablemente y que es más fácil de enseñar que C. El problema es que cada cambio sugerido debe analizarse tomando en cuenta cómo impacta en el resto de los planes de estudios. Al momento de sugerir este cambio, no se tomaron en cuenta los siguientes aspectos:

- Una UEA de Tronco General, para bien o para mal, debe servir adecuadamente a todas las ingenierías, no sólo a la mayoría.
- Las licenciaturas en Ingeniería en Computación y en Ingeniería en Electrónica dependen crucialmente de saber C.

La UEA "Algoritmos y Estructuras de Datos", seriada de forma inmediata con "Programación Estructurada" y obligatoria para Ingeniería en Computación e Ingeniería Electrónica, actualmente debe impartirse en C, C++, C# o Java. Todos estos lenguajes forman parte de la llamada *familia de C* y aunque con sus ventajas y desventajas propias, son capaces de implementar eficientemente desde aplicaciones de celular hasta sistemas de información empresarial. Lo anterior no es cierto con Python, ya que corre hasta 100 veces más lento que C e intenta ocultar el hardware subyacente a tal grado que es inviable usar Python

en "Algoritmos y Estructuras de Datos" donde temas explícitos son la eficiencia y el manejo de memoria.

¿Es deseable que un alumno sepa C y Python? Indudablemente. Lo que argumento no es la utilidad de Python como lenguaje, sino que la UEA "Programación Estructurada" de la DCBI-A no es el lugar correcto para aprenderlo. Saber lenguaje C o C++ es crucial para cualquier actividad donde la computadora deba verse como la herramienta que es y no como una caja negra; es decir, saber C o C++ es crucial al inscribir la UEA "Algoritmos y Estructuras de Datos" y la rama de UEA que depende de ella. Por ejemplo, una vez aprobada la UEA "Algoritmos y Estructuras de Datos", los alumnos de Computación y Electrónica deben cursar la UEA "Microprocesadores" donde los alumnos deben programar en lenguaje de ensamblado.

Tampoco es justificación el hecho de que muchas universidades en el mundo actualmente están cambiando a Python como primer lenguaje de programación, porque estas mismas universidades son las que cambiaron a Java como primer lenguaje de programación cuando éste se volvió popular, sólo para darse cuenta una década después que esto había sido un grave error. En estas universidades, el problema de una mala elección como primer lenguaje se atenúa un poco si sus cursos son semestrales, pues pueden dedicarse algunas clases a aprender un nuevo lenguaje o incluso se les puede pedir a los alumnos que lo aprendan de tarea durante algunas semanas, mientras se ven otros temas en clase. El ritmo trimestral de la UAM no permite hacer esto último fácilmente. Por otra parte, la UAM Azcapotzalco fue una de las instituciones que no cometió el error de usar Java como primer lenguaje y esto no fue casualidad, sino producto de una amplia discusión que terminó en que se tomara la decisión correcta de preferir el lenguaje C.

Dado que **sí** es deseable que un alumno de ingeniería aprenda Python, pero existe el argumento de que no es buena idea que sea en "Programación Estructurada", entonces se sugiere que Python sea un lenguaje de programación recomendado para la UEA "Métodos Numéricos" que también es de Tronco General. Actualmente esta UEA dice en sus objetivos que se deben aplicar las técnicas de métodos numéricos usando un lenguaje de programación, pero en las modalidades de conducción se deja abierta la elección a C (que el alumno debe saber con el programa actual) o a otro tipo de software idóneo (pero que el alumno aún no maneja) como Matlab (que es propietario y de paga) y Mathematica (que también es propietario y de paga). Actualmente algunos profesores usan C mientras que otros usan Matlab o incluso Excel. En la justificación que da el Grupo Temático de "Cómputo General" para proponer Python en la UEA "Programación Estructurada" se dice que Python está siendo cada vez más usado para realizar cómputo científico. Esto es cierto y Python podría ser una buena elección para la UEA "Métodos Numéricos", ya que este curso justamente trata temas de esta índole. Además, es más fácil aprender Python sabiendo C que viceversa.

3. Ver un ambiente de desarrollo integrado y no necesariamente UNIX.

La justificación dada por el Grupo Temático es que es ver UNIX es poco flexible (este estándar no se cumple en Windows) y que sólo las licenciaturas de Ingeniería en Computación, Ingeniería Electrónica y posiblemente Ingeniería Física necesitan aprender comandos UNIX para usarse en una terminal de línea de comandos. El problema es que una UEA de Tronco General, para bien o para mal, debe servir adecuadamente a todas las ingenierías, no sólo a la mayoría. Es **muy peligroso** usar una justificación del tipo "sólo les sirve a los alumnos de algunas ingenierías, pero no a todas" para quitar un tema de una UEA del Tronco General de la DCBI-A, porque los programas de estas UEA están diseñados para cubrir las necesidades de las ingenierías afines aún si todos los alumnos de la División deben aprobarlas. Por ejemplo, el temario de "Termodinámica" está diseñado para cubrir las necesidades de Ambiental, Física, Mecánica, Metalúrgica y Química pero los alumnos de Computación deben aprobarla con el mismo temario aún si pareciera que *no es* de su interés especial. Aplicando estrictamente la lógica de la justificación usada por el Grupo Temático hasta sus últimas consecuencias, el Tronco General podría incluso desaparecer.

Para dar una alternativa viable que no debilite el programa de la UEA, menciono que el estándar POSIX es ampliamente considerado lo mínimo esperado en sistemas de cómputo general en todo el mundo y que, a

diferencia del estándar UNIX, Windows sí es compatible con POSIX. Aspectos que todos dan por hecho (como que un *byte* es de 8 bits y no de 7 ni de 16) se garantizan en POSIX. Las UEA de Ciencias Básicas que se imparten en la UAM Azcapotzalco suelen estar siempre respaldadas por algún tipo de convención o estándar (nadie usaría una tabla periódica inventada que cambie los nombres de los elementos, por ejemplo). A diferencia de UNIX, el estándar POSIX se sigue en prácticamente todos nuestros dispositivos.

Por otra parte, es cierto que aprender a usar la línea de comandos quita algo tiempo en el curso, pero este tema es importantísimo para Computación y Electrónica. El alumno necesita aprender comandos para:

- Crear y eliminar archivos.
- Navegar a través de directorios.
- Compilar y ejecutar programas.
- Redirigir la entrada o la salida de un programa mediante archivos.

El último punto es especialmente importante, ya que la mayoría de los programas que usan archivos se limitan a leer los datos de un archivo de entrada para finalmente escribir los resultados obtenidos por el programa en un archivo de salida. Lo anterior se puede realizar usando la línea de comandos y sin la necesidad de modificar el código de un programa de tipo interactivo. En este momento se debe mencionar que para cubrir el tema de *Archivos* de la UEA “Programación Estructurada”, algunos profesores se limitan a ver sólo archivos de texto mientras que otros ven temas más complicados. Sin embargo, el manejo de archivos tiene su propia UEA en “Almacenamiento y Estructuras de Archivos” donde se ven todas las implicaciones del uso de éstos, tanto en cuestiones de funcionalidad como de rendimiento en sistemas de cómputo reales. Si el tiempo dedicado a ver la línea de comandos es motivo de preocupación, el tema de *Archivos* de “Programación Estructurada” podría limitarse a *Archivos de texto*, o bien, eliminarse a cambio de garantizar que sí se ve el tema de redirección desde la línea de comandos. Esto no es catastrófico como parece porque realmente se está eliminando redundancia en los planes de estudio. Por otro lado, el uso de la línea de comandos no se lista explícitamente ni siquiera en la UEA “Sistemas Operativos” de Ingeniería en Computación, pues esta UEA trata temas mucho más complicados. Eliminar el uso de la línea de comandos de “Programación Estructurada” no es eliminar redundancia, sino dejar un hueco.

Adicionalmente a lo anterior, sugiero un cambio menor al temario de “Programación Estructurada”. El tema de *Programación modular* siempre ha sido controversial porque no queda realmente claro qué es lo que se debe cubrir. En particular, algunos lenguajes de programación incluyen una característica llamada “módulos”, mientras que otros lenguajes como C no tienen algo similar. Hace décadas, la programación modular simplemente consistía en separar un programa en partes bien definidas. En el lenguaje C, esto sugiere que lo que se debe ver es el tema de compilación separada, el cual es un tema complicado pues implicar ver qué es el código objeto, qué es el proceso de enlazado y cuál es la visibilidad de los símbolos de un programa. Creo que ver esto en esta UEA es equivocado. Dado que el tema *Funciones y paso de parámetros* no se menciona en la lista de temas de la UEA “Programación Estructurada” pero todo mundo lo cubre, sugiero simplemente reemplazar el tema *Programación modular* por este último.

Sin más por el momento me despido, esperando que mi opinión sea tomada en consideración tanto por la Comisión como por el Grupo Temático responsable. El objetivo de todos nosotros como profesores es el mismo: que nuestros alumnos salgan mejor preparados. Espero que la discusión de éste y otros temas enriquezca nuestros programas y planes de estudio para facilitar el que se logre nuestro objetivo conjunto.