



12:15 a 13:45

- Tiempo máximo para realizar el trabajo de programación propuesto: 90 minutos
- Entrega del trabajo a través de la plataforma Moodle

En el teclado de un computador, las teclas correspondientes a letras del alfabeto generan por defecto caracteres en minúsculas, siendo necesario presionar de forma combinada la tecla «mayúsculas» con la de una letra para obtener el carácter de una letra mayúscula. La tecla de bloqueo de mayúsculas invierte el modo en el que estos caracteres se generan: cuando está activada, los caracteres correspondientes a teclas destinadas a letras se generan en mayúsculas de forma predeterminada y en minúsculas cuando la pulsación de la tecla correspondiente a una letra se combina con la tecla «mayúsculas».

Cuando se teclea un texto con la tecla de bloqueo de mayúsculas activada inadvertidamente, se genera una cadena de caracteres en el que letras mayúsculas y minúsculas están intercambiadas, como en el siguiente ejemplo:

eXAMEN PRÁCTICO DE pROGRAMACIÓN 1

Tenemos un conjunto de ficheros de texto en los que algunas líneas (pero no todas) han sido escritas, accidentalmente, con la tecla de bloqueo de mayúsculas activada. Antes de corregirlos manualmente, se ha decidido tratar de automatizar el proceso. Una posibilidad consiste en considerar que una línea en la que, simultáneamente, aparece al menos una minúscula y hay más letras mayúsculas que minúsculas ha sido escrita con el bloqueo de mayúsculas activado y debiera ser corregida, cambiando mayúsculas por minúsculas y viceversa en dicha línea.

Se pide un programa C++ que solicite al usuario el nombre de un fichero de texto que se desea corregir y el nombre del fichero corregido según el procedimiento explicado en el párrafo anterior. Cuando sea ejecutado, si no hay ningún problema con la apertura y creación de los ficheros involucrados, el programa informará de que ha podido generar el nuevo fichero correctamente:

Escriba el nombre del fichero original: <u>moby-dick.txt</u> Escriba el nombre del fichero corregido: <u>corregido.txt</u> El fichero "corregido.txt" se ha generado correctamente.

En el ejemplo anterior, si el contenido del fichero «moby-dick.txt» fuese el que aparece a continuación:

* Enunciado adaptado a los cambios hechos en el curso 2020-21



MOBY DICK
by Herman Melville

CHAPTER 1. LOOMINGS.

CALL ME iSHMAEL. SOME YEARS AGO –NEVER MIND HOW LONG PRECISELY– HAVING little or no money in my purse, and nothing particular to interest me ON SHORE, i THOUGHT i WOULD SAIL ABOUT A LITTLE AND SEE THE WATERY PART OF THE WORLD. IT IS A WAY i HAVE OF DRIVING OFF THE SPLEEN AND regulating the circulation. Whenever I find myself growing grim about THE MOUTH; WHENEVER IT IS A DAMP, DRIZZLY NOVEMBER IN MY SOUL; WHENEVER i FIND MYSELF INVOLUNTARILY PAUSING BEFORE COFFIN WAREHOUSES, AND bringing up the rear of every funeral I meet; and especially whenever my hypos get such an upper hand of me, that it requires a strong moral principle to prevent me from deliberately stepping into the street, and methodically knocking people's hats off–then, I account it high time to get to sea as soon as I can.

El programa, una vez ejecutado, habría creado el fichero «corregido.txt», con el siguiente contenido:

MOBY DICK
by Herman Melville

CHAPTER 1. LOOMINGS.

Call me Ishmael. Some years ago –never mind how long precisely– having little or no money in my purse, and nothing particular to interest me on shore, I thought I would sail about a little and see the watery part of the world. It is a way I have of driving off the spleen and regulating the circulation. Whenever I find myself growing grim about the mouth; whenever it is a damp, drizzly November in my soul; whenever I find myself involuntarily pausing before coffin warehouses, and bringing up the rear of every funeral I meet; and especially whenever my hypos get such an upper hand of me, that it requires a strong moral principle to prevent me from deliberately stepping into the street, and methodically knocking people's hats off–then, I account it high time to get to sea as soon as I can.

En el fichero de ejemplo «moby-dick.txt» aparecen en negrita las cinco únicas líneas en las que, teniendo al menos una letra minúscula, el número de mayúsculas supera al de minúsculas. En el ejemplo «corregido.txt», aparecen esas cinco líneas corregidas. En estos ejemplos, la negrita se ha añadido únicamente para facilitar la lectura y comprensión del mismo.

En los ficheros con los que se va a trabajar las letras que aparecen pertenecen exclusivamente al alfabeto inglés.

En el caso de que el programa no pueda abrir el primer fichero o no pueda crear el segundo, se limitará a escribir el mensaje de error correspondiente:

Escriba el nombre del fichero original: george-r-r-martin.txt
Escriba el nombre del fichero corregido: corr.txt
No se ha podido acceder al fichero "george-r-r-martin.txt".

* Enunciado adaptado a los cambios hechos en el curso 2020-21



Escriba el nombre del fichero a leer: moby-dick.txt
Escriba el nombre del fichero a escribir: a/b.txt
No ha podido crearse el fichero "a/b.txt".

El programa no tiene que preocuparse acerca de la ubicación de los ficheros a leer o escribir. En los ejemplos de ejecución anteriores, los ficheros involucrados deberían estar y serían creados en el directorio de ejecución del programa solicitado (en el caso de ejecutar el código con la extensión Code Runner de Visual Studio Code, en el mismo directorio en el que se encuentra el código fuente).

El desarrollo del programa solicitado en este examen se puede basar en el código personal de las prácticas o del trabajo obligatorio, aunque todo el código necesario para compilar correctamente el programa deberá entregarse en un único fichero denominado «practico.cpp» que contenga tanto la función main del programa solicitado como otras en las que pueda haberse apoyado.

En la tarea Moodle en la que hay que entregar el examen, se ha dejado el fichero «moby-dick.txt» con el ejemplo que figura en este enunciado con el objeto que pueda utilizarse para hacer pruebas (se ha dejado también en el repositorio correspondiente a la solución de este problema). También pueden realizarse pruebas con cualquier otro fichero.

Presentación del trabajo y criterios de evaluación

Cada estudiante presentará, antes de hora que se indique al comienzo del examen, a través de la plataforma Moodle el fichero con el código del módulo principal del programa, «practico.cpp». El contenido del fichero vendrá encabezado por un comentario con el nombre y apellidos del estudiante.

En este examen práctico **se valorará especialmente que el programa pueda ser compilado sin errores y que, al ser ejecutado, proporcione los resultados esperados.**

También se valorará la eficiencia de la solución presentada, su diseño, modularidad, la adecuada especificación de cada una de las funciones que integren la solución y la legibilidad del código atendiendo a los criterios de la *Guía de estilo para programar en C++* publicada en Moodle.

* Enunciado adaptado a los cambios hechos en el curso 2020-21



Criterios de corrección de los exámenes prácticos

- Compilación (establece un factor de corrección en la nota)
 - Si compila correctamente, el factor es 1.
 - Si compila con advertencias, el factor está entre 0,7 o 0,9, en función de la gravedad de las mismas.
 - Algunos errores de compilación y de ejecución tienen un factor de 0,7:
 - Presencia de cláusulas de inclusión innecesarias de bibliotecas no solicitadas
 - Errores de ejecución por modificar (anteponiendo rutas) el nombre de los ficheros introducidos por el usuario
 - Otro tipo de errores de compilación tienen un factor de corrección de 0.
- Ejecución (50%)
 - Se han realizado 5 pruebas:
 - Con el fichero «moby-dick.txt» suministrado como ejemplo.
 - Con un fichero de texto en el que no había que realizar ningún cambio en ninguna línea.
 - Con un fichero de texto en el que había que corregir todas las líneas.
 - Con la entrada «inexistente.txt», que no corresponde con ningún fichero que exista, como primer nombre de fichero.
 - Con un nombre de fichero que sí existe como primer nombre de fichero y con la entrada «a\b.txt» como segundo, no correspondiendo «a» con el nombre de ningún directorio del directorio de ejecución (se ha ejecutado en Windows 10).
 - Cuando los ficheros resultantes eran correctos, pero se ha añadido una línea en blanco de más, se ha puntuado con un 9 (o aplicado un factor de corrección de 0,9).
 - Cuando se han omitido los finales de línea, se ha puntuado con un 7 (o aplicado un factor de corrección de 0,7).
- Diseño (50%)
 - Se han valorado los siguientes aspectos:
 - Especificaciones (30%)
 - Diseño (30%)
 - Modularidad (20%)
 - Utilización de constantes simbólicas y no literales (10%)
 - Tabulación y legibilidad (10%)



Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza

Programación 1

Examen práctico. 30 de enero de 2020
