

1. Dominio: el calendario gregoriano y los feriados nacionales

Para resolver esta asignación es necesario que extienda su conocimiento sobre el calendario gregoriano. Investigue en la Web o en otras fuentes (enciclopedias, etc.) lo siguiente:

- Cuáles son los días feriados dispuestos por ley en Costa Rica¹.
- Dos de esos días feriados son 'móviles', porque dependen de la celebración de la Pascua cristiana: *Jueves* y *Viernes* santos. La Pascua cristiana depende tanto del calendario solar (equinoccio civil de Primavera boreal²) como de las fases de la luna. Puede encontrar información relacionada con el método tradicional, denominado **computus**, en Wikipedia y otras fuentes³.
- Un *día hábil* es cualquier día que no sea feriado, ni sábado, ni domingo.

2. Requerimientos funcionales adicionales

Con las propiedades del calendario **gregoriano** que Ud. investigó y resolvió en la Asignación #2, más la información que obtuvo al investigar acerca de los feriados fijos y móviles en Costa Rica, desarrolle las funciones que satisfagan los siguientes requerimientos funcionales adicionales a los enunciados en la Asignación #2:

- R7 (fecha_futura): Dada una *fecha* válida f y un número entero no-negativo n , determinar la *fecha* que está n días (naturales) en el futuro. *El resultado debe ser una fecha válida.*
- R8 (dias_entre): Dadas dos *fechas* válidas, f_1 y f_2 , sin importar si $f_1 \leq f_2$ o $f_2 \leq f_1$, determinar el número de días (naturales) entre las dos fechas. Si $f_1 = f_2$, entonces $\text{dias_entre}(f_1, f_2) = 0$. *El resultado debe ser un número entero no negativo.*
- R9 (dia_semana): Dada una *fecha* válida, determinar el día de la semana que le corresponde, con la siguiente codificación: 0 = domingo, 1 = lunes, 2 = martes, 3 = miércoles, 4 = jueves, 5 = viernes, 6 = sábado. *El resultado debe ser un número entero, conforme a la codificación indicada.*
- R10 (fecha_futura_habil): Dada una *fecha* válida f y un número entero no-negativo n , determinar la *fecha* que está n días hábiles en el futuro. *El resultado debe ser una fecha válida que corresponda a un día hábil.* Note que f puede corresponder a la fecha de un día no hábil.
- R11 (dias_habiles_entre): Dadas dos *fechas* válidas, f_1 y f_2 , sin importar si $f_1 \leq f_2$ o $f_2 \leq f_1$, determinar el número de días hábiles entre las dos fechas. Si $f_1 = f_2$, entonces $\text{dias_habiles_entre}(f_1, f_2) = 0$. *El resultado debe ser un número entero no negativo.*

Commented [IT1]: def fecha_futura (tupla, dias):

Commented [IT2]: def dias_entre (tupla, tupla):

Commented [IT3]: def dia_semana (tupla):

Commented [IT4]: def fecha_futura_habil (tupla, dias):

Commented [IT5]: def dias_habiles_entre (tupla, tupla):

3. Restricciones no funcionales y técnicas

- Diseñe y programe todas las funciones que implementen los requerimientos funcionales del apartado anterior. **No** es válido copiar directamente funciones o métodos de bibliotecas, pero sí es lícito adaptar código de otros autores, siempre que se cite la fuente y se explique *todo* lo que hace el código resultante. Todo cálculo y lógica del programa debe ser *explicado*, sea en comentarios o en documentación externa al programa. Toda programación por entregar debe ser producto de su propio trabajo o de una reutilización adecuadamente contextualizada y explicada.
- En caso de usar materiales elaborados por otros autores, debe indicar correctamente *todas* sus fuentes de información y dar referencias bibliográficas completas (use un estándar, como los de IEEE, ACM o APA). Es imprescindible citar a los autores de cualquier material que no sea propio e indicar dónde puede ser obtenido tal material.

¹ http://www.mtss.go.cr/temas-laborales/02_Feriados_ind.pdf

² El equinoccio civil de Primavera del hemisferio norte no se corresponde exactamente con el equinoccio

³ <https://es.wikipedia.org/wiki/Computus>, <https://en.wikipedia.org/wiki/Computus>, <http://code.activestate.com/recipes/576517-calculate-easter-western-given-a-year/>, <https://dzone.com/articles/algorithm-calculating-date>.

Programa: Ingeniería en Computación
Curso: Aseguramiento de la calidad del software
Asignación 3: Extender programas para implementar nuevos requerimientos funcionales
Instructor: Ignacio Trejos Zelaya

Página 2 de 3

- Ud. tiene libertad para extender los requerimientos en cuanto a la detección de casos de error y la manera en que estos serán señalados vía programación. Es decir, Ud. puede desarrollar sus soluciones 'a la defensiva' para obtener componentes más robustos.
- Resuelva todos los requerimientos en el lenguaje de programación Python (versión 3.6 o superior).
- Construya su programa de manera que el código sea legible.
- No nos interesa construir ni evaluar una interfaz de usuario amigable (en este momento).
- Pueden trabajar en grupos de hasta tres (3) miembros.
- Desarrolle su solución a los nuevos requerimientos, reutilizando, en lo posible, el programa que construyó para resolver la Asignación #2. Es válido modificar partes de su programa anterior, pero debe justificar los cambios y dejarlos bien documentados (interna y externamente al programa).
- Deben llevar cuenta de las modificaciones realizadas sobre sus artefactos y distinguir entre las versiones entregadas como parte de la Asignación #2 y las que entreguen en la Asignación #3. Incluyan un análisis somero de las diferencias entre los componentes presentes en ambas entregas.
- Lleven cuentas del esfuerzo (horas/persona) dedicado a: investigación acerca del dominio del problema, diseño de la solución, programación, pruebas, correcciones y re-trabajo, documentación, integración final de la entrega. Elaboren y adjunten una tabla que resuma esta información, con los tiempos redondeados a cuartos de hora.
- Su solución debe ser entregada en una **carpeta comprimida en formato .zip**, que comprenda: portada (que identifique a los miembros del grupo con nombres, apellidos y números de carnet), requerimientos funcionales, decisiones de diseño tomadas, código fuente de su solución, evidencias de las pruebas realizadas, análisis de resultados obtenidos, fuentes de información utilizadas, proceso seguido y tabla resumen de esfuerzos dedicados a esta asignación. El código fuente debe estar en una carpeta aparte, dentro de la misma carpeta comprimida.

4. Grupos

A continuación aparece la lista de clase con los grupos para esta Asignación (casi idénticos a los grupos de la Asignación 2).

Carné	Nombre	Asig 3 grupo
2015011460	ALFARO SIBAJA ANTHONY JOSUE	1
201235858	ALVARADO CHOU DANIEL	2
2015090756	BARRANTES BALLESTERO LUIS ANDRES	3
2015108353	BOLAÑOS RODRIGUEZ RONALD ANDRES	4
2015121963	BURGOS RETANA GEOVANNY FRANCISCO	5
2015053083	CALVO PEREZ IVAN FELIPE	6
2014160109	CORRALES GONZALEZ JENNY CRISTINA	2
2015099456	FALLAS DELGADO JIMY JOSE	7
2013033195	FERNANDEZ DIAZ JOSUE IGNACIO	5
2015049140	GONZALEZ NUÑEZ WILBERT JAFET	5
2015182286	GUEVARA MORA GREIVIN EDICSON	8
2015019659	GUZMAN VILLALTA DIEGO ARMANDO	3
2015106169	HERRERA HERNANDEZ JAKE ENRIQUE	1

Programa: Ingeniería en Computación
Curso: Aseguramiento de la calidad del software
Asignación 3: Extender programas para implementar nuevos requerimientos funcionales
Instructor: Ignacio Trejos Zelaya

Página 3 de 3

2015032254	HIDALGO BLANCO LUIS DIEGO	7
2015182213	LEANDRO MIRANDA ANTHONY FABRICIO	1
2015017726	MARTINEZ CAMACHO JONNATHAN GERARDO	9
2015071320	MOLINA ALPIZAR ALEXANDRA MARIA	10
2015096633	MONTERO SUAREZ DAVID ALEJANDRO	11
2015083527	MORA FALLAS ADAN DE JESUS	6
2015127165	MORALES RODRIGUEZ CARLOS JOSUE	12
2015183026	NAVARRO FUENTES FAURICIO JOSE	11
200565484	OBANDO RODRIGUEZ KENNETH ROBERTO	13
2015171007	PACHECO QUESADA EDGAR ALEJANDRO	13
2014071807	QUESADA PEREZ JOSE PABLO	9
2015093705	RAMIREZ SEGURA ALVARO JOSUE	12
2015008501	ROJAS SEMERARO MARIANA	4
2015105407	ROMAN ALFARO JOCELYN PAMELA	8
2015073785	SANCHO ARROYO ANDRES EMILIO	10
2015021366	TENORIO SOLIS DIEGO ANDRÉS	4
201256457	VILLALOBOS SCHMIDT JOSUE ⁴	14
2015183024	ZUÑIGA MOYA DAVID RICARDO	9

5. Entrega

- **Fecha de entrega: 2018.04.11, 23:59**
- Enviar su solución a la dirección del profesor: itrejos@itcr.ac.cr.
- Cada solución tendrá como identificación "Asignación 3 - ", concatenada con los números de carnet de los estudiantes, ordenados ascendentemente (de izquierda a derecha) y separados por un guion medio rodeado por espacios (' - ').
- El asunto de su correo debe tener el prefijo "IC-6831: ", seguido por el nombre completo del archivo como se indicó arriba.
- Esta asignación vale 10 puntos.

⁴ Vamos a procurar integrarlo a un grupo pre-existente.