

Indicaciones específicas:

- Esta evaluación contiene 7 páginas (incluyendo esta página) con 3 preguntas. El total de puntos son 20.
- El tiempo límite para la evaluación es 100 minutos.
- Cada pregunta deberá ser respondida en un solo archivo con el número de la pregunta.
 - p1.cpp
 - p2.cpp
 - p3.cpp
- Deberás subir estos archivos directamente a www.gradescope.com, uno en cada ejercicio. También puedes crear un .zip

Competencias:

- Para los alumnos de la carrera de Ciencia de la Computación
 - Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina. (Evaluar)
 - Analizar problemas e identificar y definir los requerimientos computacionales apropiados para su solución.(Usar)
 - Utilizar técnicas y herramientas actuales necesarias para la práctica de la computación. (Usar)
- Para los alumnos de las carreras de Ingeniería
 - Capacidad de aplicar conocimientos de matemáticas (nivel 3)
 - Capacidad de aplicar conocimientos de ingeniería(nivel 2)
 - Capacidad para diseñar un sistema, un componente o un proceso para satisfacer las necesidades deseadas dentro de restricciones realistas (nivel 2)

Calificación:

Tabla de puntos (sólo para uso del professor)

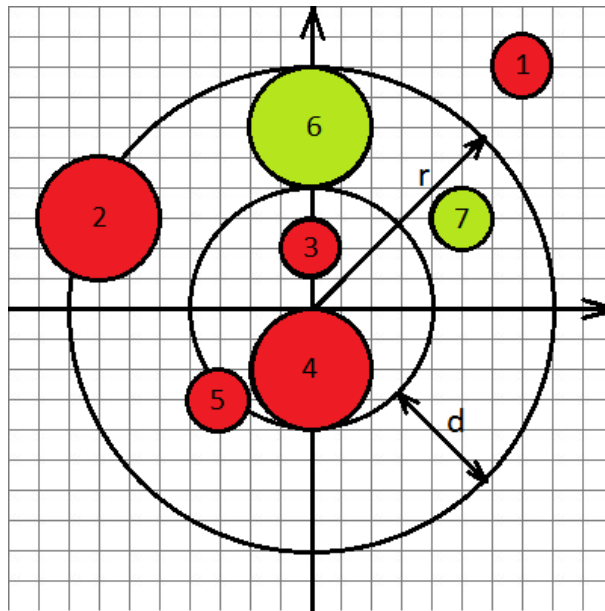
Question	Points	Score
1	7	
2	6	
3	7	
Total:	20	

1. (7 points) **Corona circular**

Calcular la cantidad de círculos pequeños que se encuentran completamente dentro de la corona circular de tamaño d . El ingreso de datos son.

- La primera cadena contiene dos números enteros r y d , el radio de círculo pequeño y el tamaño de la corona circular.
- La siguiente línea contiene un número entero n , que es el número de círculos pequeños.
- Luego cada una de las siguientes n líneas contiene tres números enteros x_i , y_i y r_i , donde x_i y y_i son las coordenadas del centro de la i -ésima del círculo pequeño, r_i es el radio de la i -ésima del círculo pequeño.

En la siguiente Figura se muestra la corona circular y círculos pequeños del Ejemplo 1.



Listing 1: Ejemplo 1

```
r d: 8 4
n: 7
c1: 7 8 1
c2: -7 3 2
c3: 0 2 1
c4: 0 -2 2
c5: -3 -3 1
c6: 0 6 2
c7: 5 3 1
cantidad de circulos: 2
```

Listing 2: Ejemplo 2

```

r d: 10 8
n: 4
c1: 0 0 9
c2: 0 0 10
c3: 1 0 1
c4: 1 0 2
cantidad de circulos: 0

```

La rúbrica para esta pregunta es:

Criterio	Excelente	Adecuado	Mínimo	Insuficiente
Ejecución	El diseño del algoritmo es ordenado y claro, siguiendo buenas prácticas en programación. La ejecución es correcta (3pts)	El diseño del algoritmo es ordenado y claro. La ejecución es correcta (2pts)	La ejecución es correcta (1pts).	La ejecución no es correcta (0.5pts)
Sintaxis	No existen errores sintácticos o de compilación (2pts)	Existen algunos errores sintácticos de menor relevancia, que no afectan el resultado (1.5pts).	Existen errores sintácticos en la forma de ejecución, que no afectan el resultado (1pts).	El código tiene errores de sintaxis que afectan el resultado (0.5pts).
Optimizacion	El código es óptimo y eficiente. De buen performance e interacción con el usuario (2pts)	El código es de buen performance durante la ejecución (1.5pts).	El código no está optimizado pero la ejecución no es deficiente (1pts).	El código no está optimizado y la ejecución es deficiente (0pts).

2. (6 points) **Año es bisiesto**

Desarrolle un programa que permita leer como dato una fecha: día, mes y año, para que luego imprima **la cantidad de días que faltan para que llegue año nuevo**. Recuerde que la cantidad de días que tiene cada mes es como se indica en la tabla:

Número de mes	Número de días
1,3,5,7,8,10,12	31
4,6,9,11	30
2	28 si no es año bisiesto y 29 si es bisiesto

Para efectos de este programa, considere que un año es bisiesto si es múltiplo de 4.

Resuelva el programa utilizando al menos: La función hallaDias, que reciba como parámetro el día, mes y año y devuelva la cantidad de días que faltan para el año nuevo. De no escribir esta función, no se asignará el puntaje a esta pregunta. Utilizar paso por referencia..

Listing 3: Ejemplo 1

```
Dia: 14
Mes: 10
Anio: 2020
Los dias que falta para anio nuevo: 78
```

Listing 4: Ejemplo 2

```
Dia: 27
Mes: 7
Anio: 2020
Los dias que falta para anio nuevo: 157
```

Listing 5: Ejemplo 3

```
Dia: 1
Mes: 1
Anio: 2020
Los dias que falta para anio nuevo: 365
```

Listing 6: Ejemplo 3

```
Dia: 31
Mes: 12
Anio: 2019
Los dias que falta para anio nuevo : 0
```

La rúbrica para esta pregunta es:

Criterio	Excelente	Adecuado	Mínimo	Insuficiente
Ejecución	El diseño del algoritmo es ordenado y claro, siguiendo buenas prácticas en programación. La ejecución es correcta (2pts)	El diseño del algoritmo es ordenado y claro. La ejecución es correcta (1.5pts)	La ejecución es correcta (1pts).	La ejecución no es correcta (0.5pts)
Sintaxis	No existen errores sintácticos o de compilación (2pts)	Existen algunos errores sintácticos de menor relevancia, que no afectan el resultado (1.5pts).	Existen errores sintácticos en la forma de ejecución, que no afectan el resultado (1pts).	El código tiene errores de sintaxis que afectan el resultado (0.5pts).
Optimizacion	El código es óptimo y eficiente. De buen performance e interacción con el usuario (2pts)	El código es de buen performance durante la ejecución (1.5pts).	El código no está optimizado pero la ejecución no es deficiente (1pts).	El código no está optimizado y la ejecución es deficiente (0pts).

3. (7 points) **Arrays dinámicos**

Escriba un programa que permita leer como dato un número entero mayor a 15, el cual representará el número de alumnos de un salón de clase. El programa luego deberá dimensionar un array dinámico para leer por teclado cada una de las N notas para luego hallar:

Se aconseja mostrar algunos ejemplos sobre la ejecución correcta del código.

- Imprimir solo las notas que están por encima del promedio.
- Imprimir la menor nota
- Imprimir el promedio eliminando la menor y mayor nota. Si el valor de la nota menor y mayor se repite se deberá eliminar todas las ocurrencias y realizar el cálculo del promedio con las notas restantes.
- Diseñe el programa de tal manera que se utilicen funciones.

Nota: Utiliza punteros y arrays dinámicos, no se puede utilizar vector.

La rúbrica para esta pregunta es:

Criterio	Excelente	Adecuado	Mínimo	Insuficiente
Ejecución	El diseño del algoritmo es ordenado y claro, siguiendo buenas prácticas en programación. La ejecución es correcta (3pts)	El diseño del algoritmo es ordenado y claro. La ejecución es correcta (2pts)	La ejecución es correcta (1pts).	La ejecución no es correcta (0.5pts)
Sintaxis	No existen errores sintácticos o de compilación (2pts)	Existen algunos errores sintácticos de menor relevancia, que no afectan el resultado (1.5pts).	Existen errores sintácticos en la forma de ejecución, que no afectan el resultado (1pts).	El código tiene errores de sintaxis que afectan el resultado (0.5pts).
Optimizacion	El código es óptimo y eficiente. De buen performance e interacción con el usuario (2pts)	El código es de buen performance durante la ejecución (1.5pts).	El código no está optimizado pero la ejecución no es deficiente (1pts).	El código no está optimizado y la ejecución es deficiente (0pts).