

GUÍA DE PROBLEMAS

1. Escriba una función para generar el N-ésimo elemento de la sucesión de Fibonacci (0, 1, 1, 2, 3, 5, 8, 13,...). El planteamiento del algoritmo correspondiente se hace a partir del análisis de la sucesión, en la que se puede observar que un tercer valor de la serie está dado por la suma de los dos valores previos. De aquí que se asignan los dos valores para sumar (0, 1), que dan la base para obtener el siguiente elemento que se busca.
2. Los libros publicados tienen un código de 10 dígitos, usualmente localizado en la contraportada. El décimo dígito es un dígito de verificación. A este dígito se le conoce como checksum (o "suma de verificación"). Esto significa que si se solicita un libro utilizando su ISBN (International Standard Book Number), la editorial puede verificar que no se haya cometido un error. Ellos simplemente verifican el dígito de checksum. A continuación se muestra cómo funciona el dígito de checksum:

Multiplique el primer dígito por 10, el segundo por nueve, y así sucesivamente, hasta multiplicar el noveno dígito por dos. Luego sume los valores. Por ejemplo, el ISBN 0-13-911991-4 da el siguiente valor: $(0 \times 10) + (1 \times 9) + (3 \times 8) + (9 \times 7) + (1 \times 6) + (1 \times 5) + (9 \times 4) + (9 \times 3) + (1 \times 2) = 172$

Luego divida el resultado por 11 y calcule el residuo. En el ejemplo, $172 \div 11 = 15$ residuo 7. Si el residuo es cero, entonces el dígito de checksum es cero. De otra manera, reste el residuo de 11 para obtener el dígito de checksum ($11 - 7 = 4$). Es posible que se llegue a obtener un valor de checksum de 10, lo cual requiere de un dígito más. Cuando esto sucede, se utiliza el carácter X. Escriba una función que permita calcular el dígito de verificación de acuerdo a la regla definida anteriormente.

3. Escriba una función que permita verificar un RUT. Para ello, se realiza el siguiente procedimiento:
 - Invertir el orden del rut
 - Tome los números y vaya multiplicando cada uno de ellos por la siguiente serie de números: 2, 3, 4, 5, 6, 7... en ese orden. Si se le acaban los números de la serie, vuelva a usarla desde el principio.
 - Una vez que haya multiplicado cada uno de los números, sume los resultados obtenidos.
 - Divida el número obtenido por 11 y obtenga el resto..
 - Al número 11, réstele el resto de la división anterior.
 - Si el número obtenido es 11, el dígito verificador es 0. Si el número obtenido es 10, el dígito verificador es K.
4. Dado el gran problema de contaminación en Santiago es necesario monitorear frecuentemente la calidad del aire. Para esto la Comisión de Descontaminación de Santiago ha comprado un set de computadores, los cuales se encargarán de recoger las mediciones y dar aviso en caso de emergencia.

Para probar los sistemas se necesita un programa en el cual se ingrese :

- El límite que indica cuando el aire es bueno o malo.
- Recibir las mediciones de la calidad y la hora en que fue tomada.

- Pedir mediciones hasta que se ingrese la de las "2400" horas
- Encender y apagar la alarma de emergencia (sólo un mensaje en pantalla).
- Al cerrar el día mostrar cuantas mediciones se registraron, cual fue el estado final y cuantas emergencias existieron junto con el total de minutos que duraron.

El programa debe implementar el siguiente diálogo :

```

Ingrese el límite aceptable de la calidad del aire : 200
Ingrese hora de la medición : 0800
Ingrese calidad del aire : 150
Ingrese hora de la medición : 1320
Ingrese calidad del aire : 210
Atención alarma encendida!!!
....
Ingrese hora de la medición : 2400
Ingrese calidad del aire : 80
- Se tomaron N medicion(es).
- Hubo X alarma(s)
- El total de tiempo en alarma fue: XXXX minutos
- El estado final fue : NORMAL

```

Nota : El estado final puede ser NORMAL o EMERGENCIA según el valor de la última medición.

5. Un programador trabaja en una empresa a honorarios. Debido a los problemas que se generaron el mes pasado por el cálculo de las horas trabajadas, este ha decidido que un programa que lleve el registro de las horas trabajadas es lo mejor. Como él tiene demasiado trabajo, ha decidido pedirles a ustedes que programen una pequeña aplicación, que permita ingresar la hora de entrada cuando él llega a trabajar, y cuando él se vaya ingrese la hora de salida. Al final de mes, el programa entregará la suma de horas trabajadas. Además preguntará cuál es el valor de la hora de trabajo del programador, con lo cual calculará el total líquido a pagar al programador por el trabajo realizado, además entregará el 10% por concepto de retenciones y el valor neto. Esto será el reporte que se utilizará para llenar la boleta de honorarios. Existen dos tarifas distintas según el horario en que trabaja: Horario normal, de 8:30 a 18:00 y Horario alta : cualquier otro.

El diálogo del programa será como sigue :

```

Ingrese hora de entrada : 0800
Ingrese hora de salida : 1300
Ingrese hora de entrada : 1530
Ingrese hora de salida : 1900
Ingrese hora de entrada : 0800
Ingrese hora de salida : 1200
Ingrese hora de entrada : -1
- Total horas trabajadas : 12,5
Ingrese valor a pagar por hora extra $ : yyyy
Ingrese valor a pagar por hora normal $ : xxxx
- Total líquido: $ 22500
- 10% retenciones: $ 2500
- Total neto: $ 25000

```

Nota : El programa termina al ingresar -1