

Práctica N°4
Modelos de Parcial

1. Escribir un programa que recorra los elementos de un vector v y encuentre los que son divisibles por un número dato x .
Reporte una copia del vector donde los elementos divisibles se reemplacen por "div".
Ejemplo: $v=(1:9)$; $w=[1\ 2\ \text{DIV}\ 4\ 5\ \text{DIV}\ 6\ 7\ \text{DIV}]$

2. Programe una función que detecte si un número entero positivo es o no un cuadrado perfecto: $\text{cuadrado}(x)$. Utilizando la función anterior como "esclavo" programe una función "maestro" que tenga como argumento una matriz cualquiera de orden " m " x " n ".

Esta función $\text{detecta}(A)$ debe detectar los cuadrados perfectos de la matriz de entrada y devolver una matriz similar pero con dichos cuadrados perfectos reemplazados por "CP".

3. Dado un vector de números v (dato) eliminar los elementos repetidos generando un nuevo vector w . con los elementos de este vector construir un nuevo vector u que contenga todos los productos posibles formados por 2 elementos distintos de w . Reportar el vector u .
4. Dada una matriz A de $n \times 2$, donde cada fila representa las coordenadas (x, y) de un punto en el plano. Determinar cuáles de dichos puntos se encuentran dentro de la zona determinada por los ejes cartesianos y la recta $y = a - b x$, con a y b positivos dados por el vector "param". Reportar las filas que representan puntos dentro de la zona.
5. Dado un vector de números llamado "base" de enteros positivos, buscar en otro vector v cualquiera de números enteros positivos la aparición de los elementos de "base". Cada vez que aparezca un elemento en v que ya exista en "base", reemplazar dicho elemento en v por el número 0. Reportar el vector w .
 $\text{base}=[2145]$ $v=[3456792385]$ $w=[3456790380]$
6. Programe una función que reconozca si un número " x " es par o impar. Llamele $\text{par}(x)$. Con dicho programa como esclavo programe una función maestro llamado "seleccion" que recorra un vector " v " #(dato) de enteros positivos y verifique si los elementos impares son números impares y si los elementos pares son números pares. El programa debe dejar inalterados los elementos que cumplan las condiciones y debe reemplazar por el valor 0 los elementos que no las cumplan.
Ejemplo: $v=[7, 12, 4, 6, 8, 5, 9, 20]$
 $\text{seleccion}(v) \rightarrow [7, 12, 0, 6, 0, 0, 9, 20]$

7. Escriba una función "mayor_que" que tenga como argumentos dos variables reales x y a . si el número x es mayor que a el programa debe retornar un 1 (verdadero), sino debe retornar un 0 (falso). Escriba un programa detecta que tenga como argumentos un vector v de dimensión libre y un escalar h . Utilizando el programa esclavo, se deberá analizar si cada elemento de v es mayor que el número h . Como salida debe generar un vector w de la misma longitud que v , que tenga 1s en las posiciones de los elementos mayores que h , y 0 en las menores o iguales.
8. Escriba un programa que detecte en un vector dato los números que llamaremos "contiguos", que resultan del producto de dos enteros seguidos, por ejemplo $6 = 2 \cdot 3$ o $56 = 7 \cdot 8$. El reporte del programa será una matriz de $n \cdot 3$ donde en la primera columna sea la posición del número contiguo en el vector y las otras dos sean los factores en que se descompone el contiguo.
Sugerencia: pruebe con los enteros cercanos a la raíz del número.
9. Programar una función cuyo desempeño sea generar un vector con elementos enteros positivos aleatorios entre 0 y 100, llamado " a ", de longitud " n " (argumento). Con los elementos de dicho vector " a " generar otro vector " b " cuyo primer elemento sea 1 y los elementos siguientes se calculan sumando y restando alternativamente los números del vector a . La cantidad de elementos del vector b deberá ser, entonces $n+1$.
10. Escriba una función "elementos" que tenga como argumentos una matriz cuadrada A y un escalar positivo b . El programa debe crear una nueva matriz c que tenga en su diagonal principal los valores de a multiplicados por b y fuera de su diagonal principal los valores de a divididos por b . La matriz c será la salida del programa.
11. Generar un programa esclavo divisible(x,v) que deberá determinar si un número " x " es divisible por alguno de los elementos de un vector dado " v ". La salida será TRUE o 1 si es divisible y FALSE o 0 si no lo es. Con ese esclavo generar una función Divisible(A) que deberá determinar si los elementos de una matriz cuadrada son divisibles por alguno de los elementos de su diagonal principal. En caso de que lo sean se deberán reemplazar por "d", si no lo son se reemplazarán por "nd".
12. *PARTE I:*
La secuencia de Collatz de un número entero se construye de la siguiente forma:
 - *Si el número es par, se lo divide por dos;
 - *Si es impar, se le multiplica tres y se le suma uno;
 - *La sucesión termina al llegar a uno.Dado un número entero positivo n , cree una función cuyo resultado sea su secuencia de Collatz

PARTE 2

Dado un vector, cree una matriz cuya primera columna sea el vector, y en las filas, a continuación de cada elemento, la secuencia de Collatz correspondiente. En caso que no coincida la longitud, completar con 0 una vez terminada la misma.

13. Un tablero de ajedrez es una grilla de ocho filas y ocho columnas, numeradas de 1 a 8. Dos de las piezas del juego de ajedrez son el alfil y la torre. El alfil se desplaza en diagonal, mientras que la torre se desplaza horizontal o verticalmente. Una pieza puede ser capturada por otra si está en una casilla a la cual la otra puede desplazarse. Escriba una función que reciba como argumento las posiciones en el tablero de un alfil y de una torre, e indique cuál pieza captura a la otra.
14. Diseña un programa que, dados cinco puntos en el plano, determine cuál de los cuatro últimos puntos es más cercano al primero. Un punto se representará con dos variables: una para la abscisa y otra para la ordenada. La distancia entre dos puntos (x_1, y_1) y (x_2, y_2) es $RAIZ[(x_1 - x_2)^2 + (y_1 - y_2)^2]$.
15. Sabiendo que en R existe un vector "letters" cuyos elementos son las letras del abecedario, indique una función que, dado un vector de números enteros positivos hasta el 26, "decodifique" el mensaje oculto, retornando un vector cuyos elementos sean las letras correspondientes. Indicar cantidad de vocales y consonantes.
16. Una de las técnicas de criptografía más rudimentarias consiste en sustituir cada uno de los caracteres por otro situado "n" posiciones más a la derecha. Si $n = 2$, por ejemplo, sustituiremos la "a" por la "c", la "c" por la "e", y así sucesivamente. El problema que aparece en las últimas n letras del alfabeto tiene fácil solución: en el ejemplo, la letra "y" se sustituirá por la "a" y la letra "z" por la "b". La sustitución debe aplicarse a las letras y a los dígitos (el 0 se sustituye con el 2, el 9 con el 1,...). Diseña un programa que lea un vector de letra / número y el valor de "n" y muestre su versión criptografiada.
17. Define una función que devuelva el número de días que tiene un año determinado. Ten en cuenta que un año es bisiesto si es divisible por 4 y no divisible por 100, excepto si es también divisible por 400, en cuyo caso es bisiesto. Ejemplos: El número de días de 2002 es 365: el número 2002 no es divisible por 4, así que no es bisiesto. El año 2004 es bisiesto y tiene 366 días: el número 2004 es divisible por 4, pero no por 100, así que es bisiesto. El año 1900 es divisible por 4, pero no es bisiesto porque es divisible por 100 y no por 400. El año 2000 si es bisiesto: el número 2000 es divisible por 4 y, aunque es divisible por 100, también lo es por 400).

18. Diseña una función que, dada una lista de números enteros, devuelva el número de "series" que hay en ella. Llamamos "serie" a todo tramo de la lista con valores idénticos.

Por ejemplo, el vector [1, 1, 8, 8, 8, 8, 0, 0, 0, 2, 10, 10] tiene 5 "series" (tener en cuenta que el 2 forma parte de una "serie" de un solo elemento).

19. *Primera Parte*

Se desea verificar si el generador de números aleatorios de R funciona bien. Para probarlo se debe escribir una función `prob(n,k)`. La función debe generar con el comando `runif()` un vector de n números enteros 1 y 10 (redondee para arriba). Con ese vector se verificará que la frecuencia de aparición de números $x \leq k$ sea la que indica la distribución uniforme.

Ejemplo:

`prob(100000,5)` aprox. 0.5

Segunda Parte

Idem anterior pero ahora se desea verificar el comando `rchisq`. Para probarlo se debe escribir una función `prob2(n,gl,k)`.

El vector v será el que surja de la aplicación directa del comando para "gl" grados de libertad. Con ese vector se verificará que la frecuencia de aparición de números $x \leq k$ sea la que indica la distribución chi cuadrado con gl grados de libertad.

Ejemplo:

`prob2(100000,5,7)` aprox 0.7793597

20. *Primera Parte*

Se pide escribir una función `"bin_to_int(b)"` que transforme un número binario "b" (de base 2) en un número entero.

Recuerde que así como un número con base decimal tiene unidades, decenas, centenas, etc. que son las potencias de 10, el número con base 2 utiliza las potencias de 2. Así el número "1 1 0 1" indica (de derecha a izquierda)

$$1 * 2^0 + 0 * 2^1 + 1 * 2^2 + 1 * 2^3 = 1*0 + 0*2 + 1*4 + 1*8 = 1 + 0 + 4 + 8 = 13$$

Para simplificar el número binario vendrá indicado como un vector con un elemento por número.

Ejemplo:

`b=c(1,1,0,1)` `bin_to_int(b)=13`

`b=c(1,0,1,0,0,1,1)` `bin_to_int(b)=83`

`b=c(1,1,0,1,1,0,1)` `bin_to_int(b)=109`

Segunda Parte

Idem anterior pero ahora la función se llamará `"int_to_bin(n)"` y deberá transformar un número entero positivo en un número binario. Se adjunta el link para un video explicativo de la metodología, que usted deberá programar en su función

<https://www.youtube.com/watch?v=-4rUKINeCEs>

Ejemplos: Los mismos de la parte I pero invertidos

21. En el programa R, para eliminar un elemento se pone un signo menos en la posición del elemento a eliminar. Por ejemplo $v = [3 \ 2 \ 2 \ 14]$ $w = v[-c(3)] = [3 \ 2 \ 14]$
Dado un vector de números enteros positivos v (dato) eliminar los elementos repetidos generando un nuevo vector w .
Note que cada vez que se elimina un elemento la longitud de vector cambia, por lo tanto se sugiere reemplazar en un primer paso todos los repetidos por 0 y luego eliminar los 0s.
 $v = [3 \ 2 \ 2 \ 14 \ 3]$ $w = [3 \ 2 \ 14]$
22. Escriba un programa "cercano" cuyo argumento sea una matriz de n filas y 2 columnas que representen coordenadas x e y en el plano cartesiano. El otro argumento del programa será un par ordenado v representando las coordenadas de un punto en el plano.
El programa debe detectar cuál de los puntos de la matriz está más cercano al punto v . Para calcular las distancias recuerde la fórmula: $\text{puntos}=(x1, y1)$ y $(x2, y2)$
 $\text{distancia}=\text{RAIZ CUADRADA}[(x1 - x2)^2 + (y1 - y2)^2]$.
Para encontrar la mínima distancia de las calculadas pruebe los comandos `min()` y `which.min()`
La salida debe ser un vector que contenga el j indicando la fila en la matriz A del punto más cercano, los valores x e y del punto y la distancia de dicho punto al punto dato v .
23. Escriba un programa "espejo" que genere a partir de una matriz cualquiera a otra matriz cuyos elementos sean una imagen a espejo de la original. La imagen a espejo será respecto de un eje vertical si el argumento `eje = 1` y respecto de un eje horizontal si el argumento `eje = 2`. Si encuentra un comando para hacerlo, no lo use.
24. Escribir un programa "diagonales" que extraiga de una matriz A cuadrada cualquiera los vectores formados por sus diagonales secundarias y por la contradiagonal. Si encuentra un comando para hacerlo, no lo use.
25. Escribir un programa "índices" que extraiga de una matriz todos los elementos que tengan la suma de sus dos índices (de fila y columna) múltiplos de a . El reporte deberá ser un vector indicando fila y columna para cada valor extraído.
26. Escribir un programa "intervalo" que tenga como argumento una matriz A de enteros positivos de dimensión cualquiera. El programa deberá encontrar los elementos de la matriz que son mayores e iguales que un número a y menores o iguales que un número b , ambos argumentos del mismo. El reporte deberá ser un vector indicando para cada valor encontrado, fila y columna. Si hay valores repetidos incluir todos.

27. *Primera Parte*

Se pide escribir una función "multiplo(b,k)" que verifique si un número x y los números que surgen del mismo son múltiplos de un número "k" dado (entero positivo).

Por números que surgen de x se entiende los números que se forman tomando los dígitos de x en forma secuencial y acumulativa de izquierda a derecha.

Por ejemplo del número 3948743 surgen los números: 3, 39, 394, 3948, 39487, 394874 y 3948743

Para encontrar dichos números hay 2 alternativas:

Dividir x por las potencias crecientes de x comenzando de la potencia 0 y tomar el redondeo del resultado hacia abajo

$$\text{floor}(3948743/1) = 3948743$$

$$\text{floor}(3948743/10) = 394874$$

$$\text{floor}(3948743/100) = 39487$$

...

$$\text{floor}(3948743/1000000) = 3$$

$$\text{floor}(3948743/10000000) = 0 \quad -> \text{este no se usa}$$

Otra posibilidad es ingresar el número como vector de dígitos

$$x = c(3, 9, 4, 8, 7, 4, 3)$$

Ejemplo:

$$\text{multiplo}(3948743,2) = c(394874, 3948, 394)$$

Segunda Parte

Idem anterior pero ahora la función se llamará "multiplo2(b,k)" y deberá considerar como números que surgen de x a los números que se forman tomando los dígitos de x en forma secuencial y acumulativa de izquierda a derecha y de derecha a izquierda.

Por ejemplo para el número 3948743 surgen los números:

3, 39, 394, 3948, 39487, 394874, 3948743, 3, 43, 743, 8743, 48743, 948743, 3948743 (no importa si aparecen repetidos)

Ejemplos:

$$\text{multiplo2}(3948743,4) = c(3948)$$