**Fundamentos de Programación (grupos A y C)**

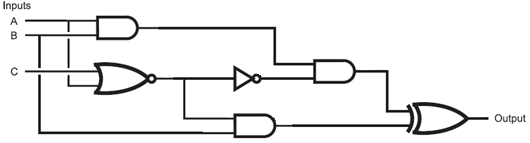
**(Grados en Ingeniería Mecánica, Electrónica Industrial y Química Industrial)**

**11 de septiembre de 2014**

***Ejercicios 1 y 2* (5*.0 p/ejercicio*)*:*** construir un programa en ***C*** ó ***Java*** lo más modular posible (atendiendo a los criterios de modularidad) y documentar el diseño preliminar con la definición de nuevas tipologías de datos, el diagrama de módulos (estructura del programa) y las interfaces de los módulos, y el diseño detallado con las definiciones de los respectivos sub-programas.

**Ejercicio 1:** construir un programa para obtener e imprimir en pantalla la función de respuesta de un circuito lógico combinatorio dado a todas las combinaciones booleanas posibles de sus entradas (tabla de verdad). Ejemplo:

|  |  |  |  |
| --- | --- | --- | --- |
| **Inputs** | | | **Output** |
| **A** | **B** | **C** |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

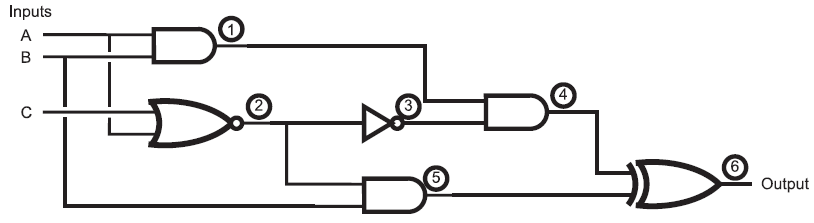


Un circuito lógico combinatorio está formado por: 1) entradas lógicas (cada una admite una señal digital binaria: 0/1), y 2) puertas lógicas elementales conectadas pasando la salida de una a la entrada de otra, con la única restricción de que los flujos de datos en una dirección y las salidas de las puertas lógicas no se pueden usar como entradas a puertas cuya salida vuelva a ellas mismas.

El programa leerá por teclado la especificación del circuito lógico combinatorio, solicitando inicialmente al usuario el nº de entradas al circuito (máximo ***10*** entradas) y el nº de puertas lógicas elementales que forman el circuito (máximo ***100***); a continuación se pedirá para cada puerta lógica su tipo (uno de los ***6*** tipos básicos), así como sus entradas (éstas pueden ser alguna de las entradas iniciales o alguna de las salidas de las puertas lógicas que forman el circuito). Una vez validado el circuito combinatorio, se procederá a calcular e imprimir en pantalla la tabla de verdad del circuito, evaluando la salida del mismo para cada una de las combinaciones booleanas de las entradas.

Sugerencia: identifique mediante un número cada puerta lógica elemental del circuito (y su salida) y construya una tabla con toda esa información. Utilice números en lugar de letras para identificar las entradas.

|  |  |  |  |
| --- | --- | --- | --- |
| **Puerta lógica** | **Tipo** | **Entrada 1** | **Entrada 2** |
| 1 | AND | A | B |
| 2 | NOR | A | C |
| 3 | NOT | 2 |  |
| 4 | AND | 1 | 3 |
| 5 | AND | B | 2 |
| 6 | XOR | 4 | 5 |



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Entrada digital** | | **Tabla de verdad de las puertas lógicas elementales** | | | | | |
|  |  |  |  |  |  |
| **A** | **B** | **AND** | **OR** | **NOT** | **NAND** | **NOR** | **XOR** |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |

**Ejercicio 2:** Construir un programa que calcule e imprima en pantalla la descomposición o factorización de una matriz rectangular ***A*** de ***m*** filas y ***n*** columnas como producto de una matriz ortogonal ***Q*** ***m***x***m*** por una matriz triangular superior ***R*** ***m***x***n*** (descomposición ***QR***): con

El programa leerá inicialmente por teclado las dimensiones ***m***y ***n*** de la matriz ***A*** (máximo ***10*** filas x ***10*** columnas), y seguidamente solicitará por teclado los elementos de dicha matriz, de ***1*** en ***1***. A continuación realizará la descomposición ***QR*** de la matriz ***A*** mediante el uso de reflexiones de Householder, y finalmente imprimirá en pantalla las dos matrices obtenidas (si la descomposición ha sido posible).

***Nota***: Una reflexión de Householder es una transformación que refleja el espacio con respecto a un plano ***π*** determinado. Si ***v*** es un vector columna unitario normal al plano ***π***, la matriz ***H*** asociada a dicha transformación se obtiene mediante la siguiente ecuación:

Es posible elegir la matriz ***H*** de manera que un vector dado no nulo ***x*** quede con una única componente no nula tras ser transformado (pre-multiplicado por ***H***), esto es, es posible reflejar dicho vector respecto de un plano de manera que el reflejo quede sobre uno de los ejes de la base cartesiana. Esta propiedad se usa para transformar gradualmente los vectores columna de la matriz ***A***=(***a1***|***a2***|…|***an***) en una matriz triangular superior. Método para la descomposición ***QR***:

1. Considerar la primera columna:

y el vector unitario:

Nota: si durante este proceso se obtiene un vector ***x*** nulo (todos sus componentes iguales a cero), detener el proceso dado (no es posible la descomposición ***QR***).

1. Elegir el plano de reflexión y formar la matriz de Householder asociada:

Vector normal al plano de reflexión:

Vector unitario:

Matriz de Householder asociada:  **🡪**

1. Pre-multiplicar la matriz ***A*** por ***H1***, obteniendo una matriz con ceros en la primera columna excepto el elemento de la primera fila:
2. Repetir ***t*** veces los pasos anteriores (***t=min(m-1,n)***), con la matriz ***A'*** que se obtiene a partir de ***H1.A*** eliminando sucesivamente la primera fila y columna, obteniéndose las matrices de Householder ***H2***, ***H3***,…, ***Ht*** (ver resumen de pasos).
3. Formar progresivamente las matrices ***Q*** y ***R***, expandiendo previamente las matrices de Householder:

Expandir hacia arriba y hacia la izquierda la matriz de Householder, completando con ***1*** en la diagonal:

Matriz ortogonal:

Matriz triangular superior:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Resumen de pasos** | | | | | |
| 1 | ***A*** | ***H1*** | ***Q1*** | ***Q 🡨 Q1*** | ***R 🡨 Q1 .A*** |
| 2 | ***A'2*** 🡪 eliminar 1ª fila y columna de ***R*** | ***H2*** | ***Q2*** | ***Q 🡨 Q1 .Q2*** | ***R 🡨 Q2 .Q1 .A*** |
| 3 | ***A'3*** 🡪 eliminar dos 1ªs filas y columnas de ***R*** | ***H3*** | ***Q3*** | ***Q 🡨 Q1 .Q2 .Q3*** | ***R 🡨 Q3 .Q2 .Q1 .A*** |
| … | … | … | … | … | … |
| t | ***A't*** 🡪 eliminar ***t-1*** 1ªs filas y columnas de ***R*** | ***Ht*** | ***Qt*** | ***Q 🡨 Q1 .Q2 …Qt*** | ***R 🡨 Qt …Q2 .Q1 .A*** |

Ejemplos:

m=n=3

m=4 n=3