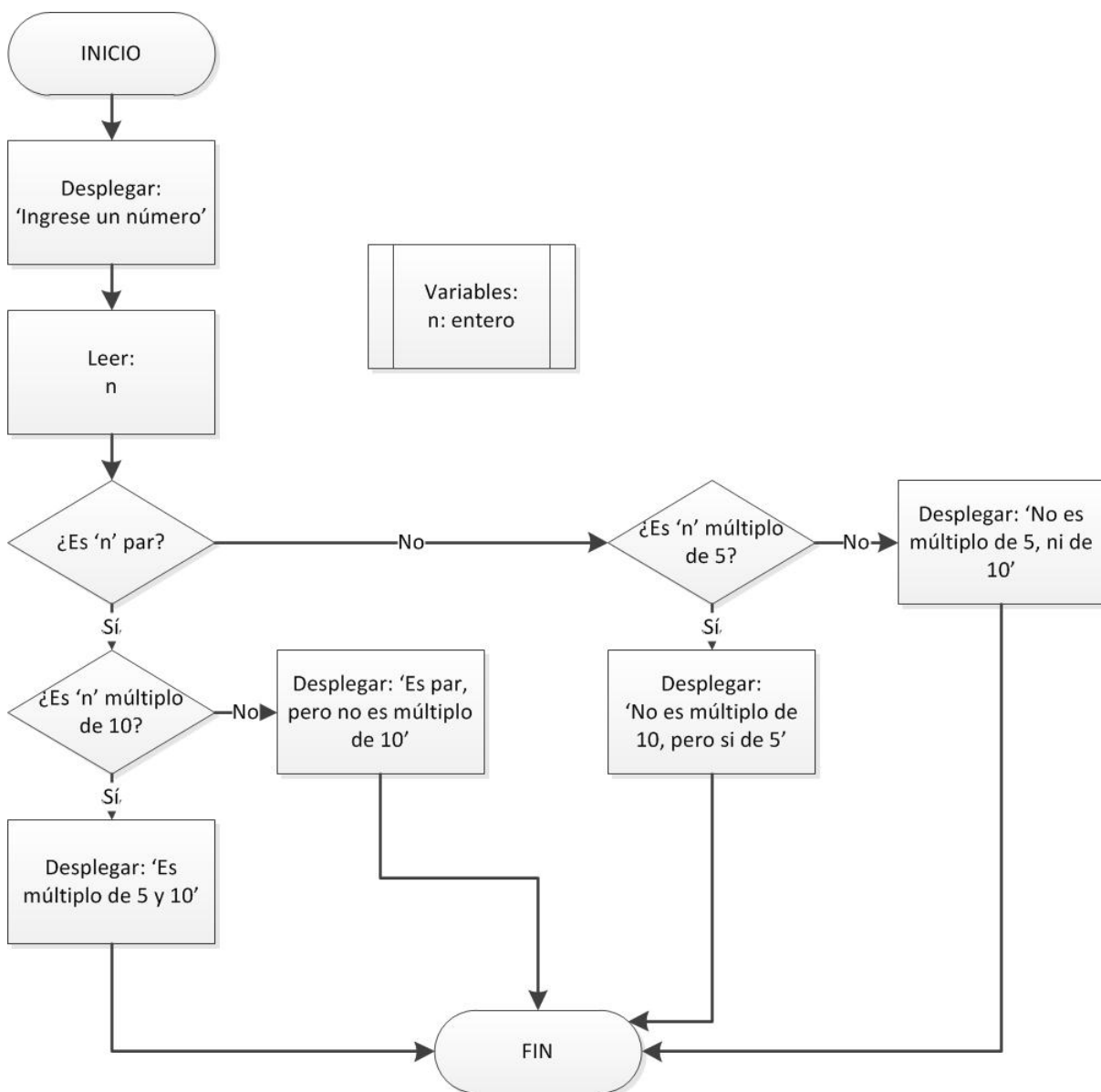


Introducción a la Programación de Computadoras (0769) - Sección N
Primer examen corto - Segundo semestre 2014 - 06/08/2014
Profesor: Iván René Morales

Instrucciones: Responda las preguntas de forma concisa utilizando lapicero negro o azul. No se permiten tachones, ni el uso de corrector. No se permite ingresar material de apoyo. Cualquier intento de fraude o copia anula automáticamente el examen a todas las partes involucradas. Duración del examen: 45 minutos.

Nombre: _____ Carnet: _____

1. (40 points) Basándose en el diagrama de flujo que se muestra a continuación, escriba un código que realice un funcionamiento equivalente, utilizando lenguaje C. Recuerde escribir todo el *esqueleto* del código. Tome en cuenta que se evaluará también la indentación (tabulaciones). Utilice la parte de posterior de esta página para escribir su código.



Solución

```
1  #include<stdio.h>
2
3  int n;
4
5  int main(void){
6      printf("Ingrese un numero: ");
7      scanf("%d", &n);
8      if (n%2 == 0){ //Verificando si 'n' es par
9          if (n%10 == 0){ //Es multiplo de 10?
10             printf("Es multiplo de 5 y 10");
11         }else{
12             printf("Es par, pero no es multiplo de 10");
13         }
14     }else{
15         if (n%5 == 0){
16             printf("No es multiplo de 10, pero si de 5");
17         }else{
18             printf("No es multiplo de 5, ni de 10");
19         }
20     }
21
22     return 0;
23 }
```

2. (50 points) Deberá resolver el siguiente problema utilizando lenguaje C.

Se desea crear un esquema de codificación para transmitir códigos secretos. Debe pedirse al usuario ingresar un número entero entre -32000 y +32000 y el programa deberá responder con un mensaje mostrando este valor luego de haber sido encriptado. Todos los resultados deben presentarse únicamente con **2 puntos decimales**. Asumiendo que x es la entrada, el algoritmo de encriptación debe seguir los siguientes criterios:

- Si $x < -5000$, entonces:
Obtener el residuo de x respecto a 20. Luego, restarle a ese resultado el número ingresado (x). Después, realizar una división entera respecto a 37 y sumarle el producto de 2 con x . A este resultado obtenido se le llama y . Finalmente, realizar la división con punto decimal entre y y x para obtener el número codificado.
- Si $-5000 \leq x \leq +5000$, entonces:
Obtener el residuo de x respecto a 5. Luego, restarle a ese resultado el número ingresado (x). Después, realizar una división entera respecto a 28 y sumarle el producto de 3 con x . A este resultado obtenido se le llama y . Finalmente, realizar la división con punto decimal entre x y y para obtener el número codificado.

- Si $x > +5000$, entonces:
Obtener el residuo de x respecto a 10. Luego, restarle a ese resultado el número ingresado (x). Después, realizar una división entera respecto a 12 y sumarle el producto de 4 con x . A este resultado obtenido se le llama y . Finalmente, realizar la división con punto decimal entre y y x para obtener el número codificado.

Solución

```
1  #include<stdio.h>
2  int x, y;
3  float resultado;
4  int main(void){
5      printf("Ingrese un numero entre -32000 y 32000");
6      scanf("%d", &n);
7      if(x < -5000){
8          y = x % 20;
9          y = y - x;
10         y = y / 37 + 2*x;
11     }else if((x >= -5000) && (x <= 5000)){
12         y = x % 5;
13         y = y - x;
14         y = y / 28 + 3*x;
15     }else{
16         y = x % 10;
17         y = y - x;
18         y = y / 12 + 4*x;
19     }
20     resultado = (float)y / x;
21     printf("Resultado: %1.2f", resultado);
22     return 0;
23 }
```

3. (10 points) Ahora escriba el comando para compilar su código fuente (piense que su archivo se llama *examenCorto.c*) y crear un archivo ejecutable llamado *examenCorto*.

gcc examenCorto.c -o examenCorto