

**Universidad Tecnológica de Panamá**  
**Facultad de Ingeniería en Sistemas Computacionales**  
**Asignatura: Desarrollo Lógico y Algoritmo**  
**Examen Parcial1**

Profesor: Napoleón Ibarra Valor: 100 puntos

Estudiante: Cédula:

Fecha Inicio: 16/09/2025 → 3:20 PM

Fecha Entrega: 16/09/2025 → 5:45 PM

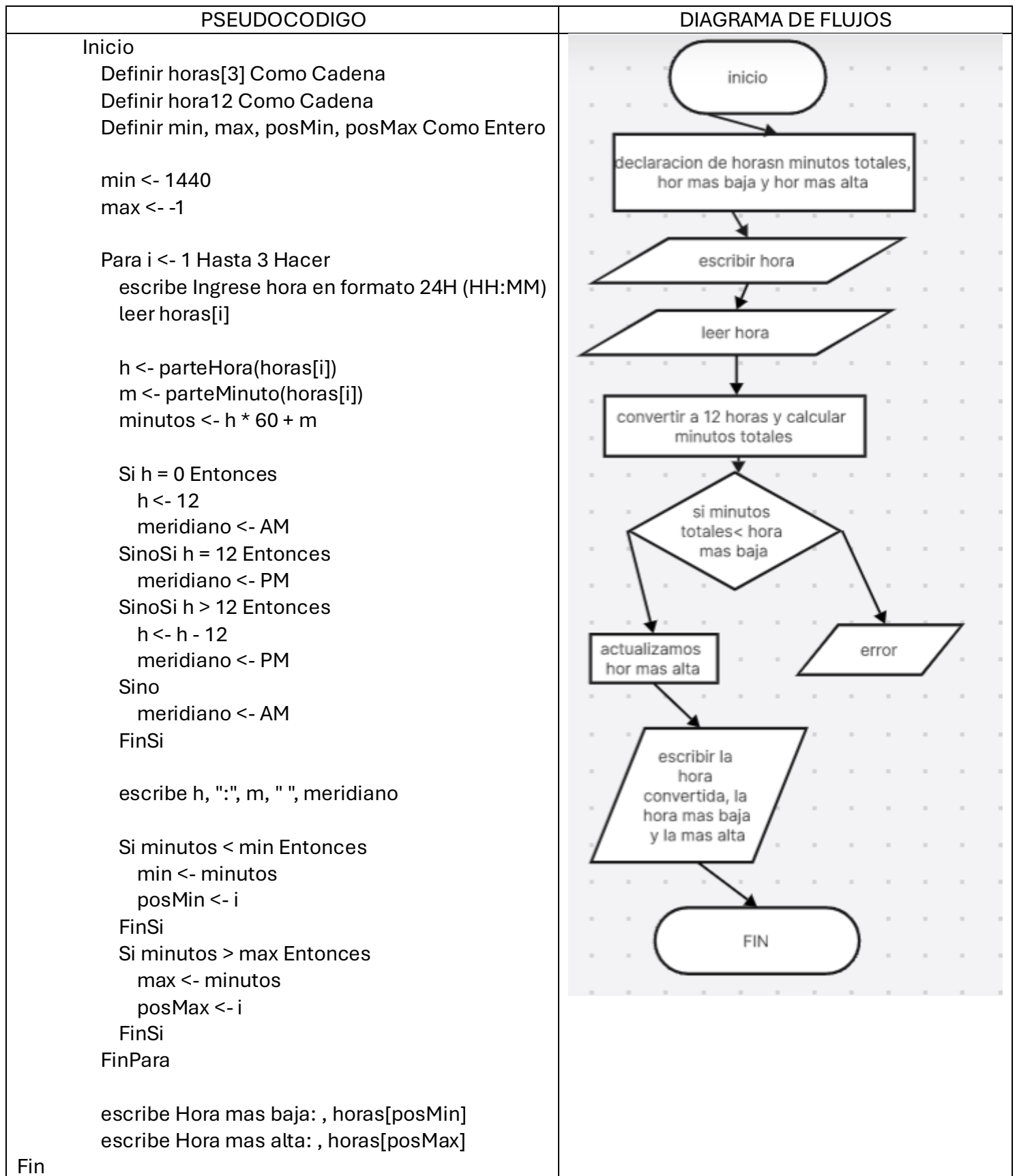
**Procedimiento:**

1. De manera individual, realizar la asignación. Una vez culminada entregue en la Plataforma Web (Team).
2. Cada uno de los códigos desarrollados debe ser presentado de manera digital, sustentado (proyector) en el aula de clases.
3. Utilizando la herramienta Internet, investigue y complemente los conceptos para el desarrollo del caso de estudio.

**I Parte. Problemas.**

- 1) Escribir una solución que lea la hora (3) en notación de 24 horas y que imprima en notación de 12. Ejemplo: Si la entrada es 13:45, la salida será 1:45 PM. El prototipo de desarrollo debe ser capaz de solicitar al usuario final que introduzca de forma exacta 5 caracteres para especificar 1 hora. Ejemplo: las 9 en punto se debe introducir: 09:00. Esto se debe realizar para 3 valores introducidos por teclado, al final me debe decir también, cuál es la hora más baja y cual es la más alta.

--	--



```

#include <stdio.h>
#include <stdlib.h>

/* run this program using the console pauser or add your own getch, system("pause") or input loop */

int main(int argc, char *argv[]) {

    char horas[3][6];
    int h[3], n[3];
    int i;

    for (i = 0; i < 3; i++) {
        printf("Introduce la hora %d en formato 24h (HH:MM): ", i+1);
        scanf("%5s", horas[i]); // Leer máximo 5 caracteres
        sscanf(horas[i], "%d:%d", &h[i], &n[i]); // Separar en hora y minutos
    }

    printf("\nHoras convertidas a formato de 12 horas:\n");
    for (i = 0; i < 3; i++) {
        int hora12 = h[i];
        char period[3] = "AM";

        if (hora12 == 0) {
            hora12 = 12;
        } else if (hora12 == 12) {
            strcpy(period, "PM");
        } else if (hora12 > 12) {
            hora12 -= 12;
            strcpy(period, "PM");
        }

        printf("%d:%02d %s\n", hora12, n[i], period);
    }

    int menor = 0, mayor = 0;
    for (i = 1; i < 3; i++) {
        if ((h[i] < h[menor]) || (h[i] == h[menor] && n[i] < n[menor])) {
            menor = i;
        }
        if ((h[i] > h[mayor]) || (h[i] == h[mayor] && n[i] > n[mayor])) {
            mayor = i;
        }
    }


    printf("\nLa hora mas baja fue: %s\n", horas[menor]);
    printf("La hora mas alta fue: %s\n", horas[mayor]);

    return 0;
}

```

2) Crear una solución que acepte fechas escritas de modo usual y que visualice en 3 números; por ejemplo: la entrada 15, febrero, 1989 debe producir la salida: 15 2 1989. Esto se debe

realizar para 3 valores introducidos por teclado, al final me debe decir también, cuál es la fecha más baja y cuál es la fecha más alta.

PSEUDOCODIGO	DIAGRAMA DE FLUJOS
<p>Inicio</p> <p>Definir dia[3], anio[3] Como Entero  Definir mesTexto[3] Como Cadena  Definir mesNum[3] Como Entero  Definir i, valor, min, max, posMin, posMax Como Entero</p> <p>min &lt;- 99999999  max &lt;- -1</p> <p>Para i &lt;- 1 Hasta 3 Hacer  Repetir  escribe Ingrese dia, mes (texto), anio separados por comas (ej: 15, febrero, 1989)  leer dia[i], mesTexto[i], anio[i]</p> <p>mesNum[i] &lt;- convertirMes(mesTexto[i]) // enero=1, febrero=2, etc.</p> <p>Si mesNum[i] &gt;= 1 Y mesNum[i] &lt;= 12 Entonces  salir del Repetir  Sino  escribe Mes invalido, intente de nuevo  FinSi</p> <p>Hasta que mesNum[i] valido</p> <p>escribe dia[i], " ", mesNum[i], " ", anio[i]</p> <p>valor &lt;- anio[i]*10000 + mesNum[i]*100 + dia[i]</p> <p>Si valor &lt; min Entonces  min &lt;- valor  posMin &lt;- i  FinSi</p> <p>Si valor &gt; max Entonces  max &lt;- valor</p>	 <pre> graph TD     Inicio([Inicio]) --&gt; Decl[declaracion de fechas, fecha min y fecha max]     Decl --&gt; Escribir[escribir dia, mes y año]     Escribir --&gt; Leer[/leer dia mes y año/]     Leer --&gt; Convert[convertir mes de texto a numero y asignar un valor numerico para comparacion]     Convert --&gt; Dec{fecha actual &gt; fecha max}     Dec -- Si --&gt; Actual[actualizamos la fecha max]     Dec -- no --&gt; Error[/error/]     Actual --&gt; EscribirFecha[/escribir la fecha en formato numerico la mas baja y la mas alta/]     EscribirFecha --&gt; Fin([FIN])   </pre>

```

posMax <- i
FinSi
FinPara

```

```

    escribe Fecha mas baja: , dia[posMin], " ",
mesNum[posMin], " ", anio[posMin]
    escribe Fecha mas alta: , dia[posMax], " ",
mesNum[posMax], " ", anio[posMax]
Fin

```

```

#include <stdio.h>
#include <string.h>
#include <ctype.h>

// Función para convertir nombre del mes a número
int mesNumero(char mes[]) {
    const char *meses[] = {"enero", "febrero", "marzo", "abril", "mayo", "junio",
        "julio", "agosto", "septiembre", "octubre", "noviembre", "diciembre"};
    int i;
    for (i = 0; i < 12; i++) {
        if (strcmp(mes, meses[i]) == 0) return i + 1;
    }
    return 0;
}

// Convierte a minúsculas y elimina espacios al inicio y final
void normalizar(char *s) {
    int i, j;
    for (i = 0; s[i]; i++) s[i] = (char)tolower((unsigned char)s[i]);
    // quitar espacios inicio
    i = 0;
    while (s[i] && isspace((unsigned char)s[i])) i++;
    if (i > 0) memmove(s, s + i, strlen(s + i) + 1);
    // quitar espacios final
    j = strlen(s);
    while (j > 0 && isspace((unsigned char)s[j - 1])) { s[j - 1] = '\0'; j--; }
}

int main() {
    int dia[3], anio[3], mesNum[3];
    char mesTexto[40];
    int i, valor;
    int min = 99999999, max = -1, minPos = 0, maxPos = 0;

    printf("Ingrese 3 fechas (dia, mesTexto, anio) separadas por comas.\n");

    for (i = 0; i < 3; i++) {
        while (1) {
            printf("Fecha %d: ", i + 1);
            if (scanf("%d, %39[^,], %d", &dia[i], mesTexto, &anio[i]) != 3) {
                printf("Formato invalido, intente de nuevo.\n");
                // limpiar buffer
                int c; while ((c = getchar()) != '\n' && c != EOF);
                continue;
            }
            normalizar(mesTexto);
            mesNum[i] = mesNumero(mesTexto);
            if (mesNum[i] >= 1 && mesNum[i] <= 12) break;
            printf("Mes invalido, intente de nuevo.\n");
        }

        printf("-> %d %d %d\n", dia[i], mesNum[i], anio[i]);

        valor = anio[i]*10000 + mesNum[i]*100 + dia[i];

        if (valor < min) { min = valor; minPos = i; }
        if (valor > max) { max = valor; maxPos = i; }
    }

    printf("\nFecha mas baja: %d %d %d\n", dia[minPos], mesNum[minPos], anio[minPos]);
    printf("Fecha mas alta: %d %d %d\n", dia[maxPos], mesNum[maxPos], anio[maxPos]);

    return 0;
}

```

3) Dadas 2 fechas en formato día (1 a 31), mes (1 a 12) y año (entero de 4 dígitos), correspondientes a la fecha de nacimiento y fecha actual, de forma respectiva. Confeccione una solución que deduzca y visualice la edad del individuo; si es la fecha de un bebe de menos de un año, la edad se debe dar en meses y días; en caso contrario, en años.

PSEUDOCODIGO	DIAGRAMA DE FLUJOS
<p>Inicio</p> <p>Definir diaN, mesN, anioN Como Entero</p> <p>Definir diaA, mesA, anioA Como Entero</p> <p>Definir edadAnios, edadMeses, edadDias Como Entero</p> <p>escribe Ingrese fecha de nacimiento (dia mes anio):</p> <p>leer diaN, mesN, anioN</p> <p>escribe Ingrese fecha actual (dia mes anio):</p> <p>leer diaA, mesA, anioA</p> <p>// Ajuste de días</p> <p>Si diaA &lt; diaN Entonces</p> <p>    mesA &lt;- mesA - 1</p> <p>    Si mesA &lt; 1 Entonces</p> <p>        mesA &lt;- mesA + 12</p> <p>        anioA &lt;- anioA - 1</p> <p>    FinSi</p> <p>    diaA &lt;- diaA + diasDelMes(mesA, anioA)</p> <p>FinSi</p> <p>// Ajuste de meses</p> <p>Si mesA &lt; mesN Entonces</p> <p>    mesA &lt;- mesA + 12</p> <p>    anioA &lt;- anioA - 1</p> <p>FinSi</p> <p>edadAnios &lt;- anioA - anioN</p> <p>edadMeses &lt;- mesA - mesN</p>	<pre> graph TD     Inicio([inicio]) --&gt; Decl[declaracion fecha de nacimiento y fecha actual]     Decl --&gt; Escribir[/escribir fecha de nacimiento y fecha actual/]     Escribir --&gt; Leer[/leer fecha de nacimiento y fecha actual/]     Leer --&gt; Calcular[calcular la diferencia de años, meses y dias]     Calcular --&gt; Decidir{la edad es menor de 1 año?}     Decidir -- Si --&gt; ImprimirMeses[/imprimir la edad en meses y dias/]     Decidir -- no --&gt; ImprimirAños[/imprimir la edad en años/]     ImprimirMeses --&gt; Fin([FIN])     ImprimirAños --&gt; Fin     </pre>

```
edadDias <- diaA - diaN
```

```
// Mostrar edad según corresponda
```

```
Si edadAnios >= 1 Entonces
```

```
    escribe Edad: , edadAnios, " años"
```

```
Sino
```

```
    escribe Edad: , edadMeses, " meses y ",  
edadDias, " dias"
```

```
Fin
```

```
#include <stdio.h>

int esBisiesto(int anio) {
    return (anio % 4 == 0 && (anio % 100 != 0 || anio % 400 == 0));
}

int diasDelMes(int mes, int anio) {
    int diasMes[] = {31,28,31,30,31,30,31,31,30,31,30,31};
    if (mes == 2 && esBisiesto(anio)) return 29;
    return diasMes[mes-1];
}

void limpiarBuffer() {
    int c;
    while ((c = getchar()) != '\n' && c != EOF);
}

int main() {
    int diaN, mesN, anioN;
    int diaA, mesA, anioA;
    int edadAnios, edadMeses, edadDias;

    printf("Ingrese fecha de nacimiento (dia mes anio): ");
    if (scanf("%d %d %d", &diaN, &mesN, &anioN) != 3) return 1;
    limpiarBuffer();

    printf("Ingrese fecha actual (dia mes anio): ");
    if (scanf("%d %d %d", &diaA, &mesA, &anioA) != 3) return 1;
    limpiarBuffer();

    if (diaA < diaN) {
        mesA -= 1;
        if (mesA < 1) {
            mesA += 12;
            anioA -= 1;
        }
        int mesAnterior = mesA;
        if (mesAnterior < 1) mesAnterior = 12;
        int anioParaDias = (mesA < 1) ? anioA - 1 : anioA;
        diaA += diasDelMes(mesAnterior, anioParaDias);
    }

    if (mesA < mesN) {
        mesA += 12;
        anioA -= 1;
    }

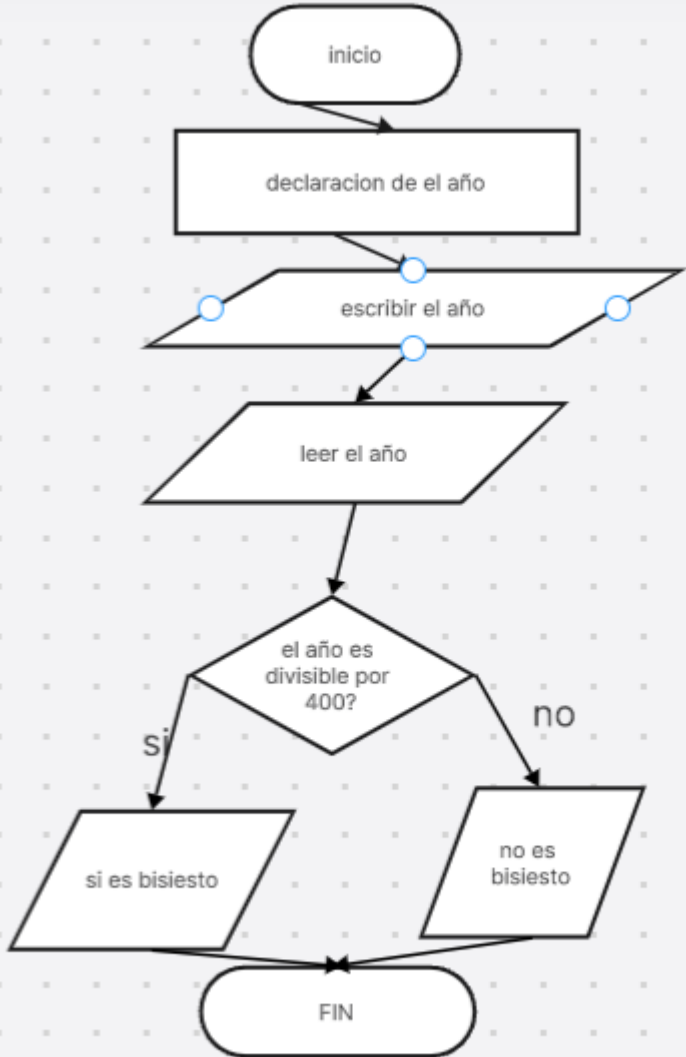
    edadAnios = anioA - anioN;
    edadMeses = mesA - mesN;
    edadDias = diaA - diaN;

    if (edadAnios >= 1) {
        printf("Edad: %d\n", edadAnios);
    } else {
        printf("Edad: %d meses y %d dias\n", edadMeses, edadDias);
    }

    return 0;
}
```



4) Codificar una solución que determine si un año es bisiesto; esto se presenta cuando es múltiplo de 4, por ejemplo, 1984; si embargo, los años que son múltiplos de 100 sólo son bisiestos cuando también son múltiplos de 400; por ejemplo, 1800 no es bisiesto, mientras que 2000, si lo es.

PSEUDOCODIGO	DIAGRAMA DE FLUJO
<p>Inicio</p> <p>Definir año Como Entero</p> <p>escribe Ingrese un año: leer año</p> <p>Si (año mod 4 = 0) Entonces</p> <p>    Si (año mod 100 = 0) Entonces</p> <p>        Si (año mod 400 = 0) Entonces</p> <p>            escribe El año , es bisiesto</p> <p>        Sino</p> <p>            escribe El año , no es bisiesto</p> <p>    FinSi</p> <p>    Sino</p> <p>        escribe El año , es bisiesto</p> <p>    FinSi</p> <p>    Sino</p> <p>        escribe El año , no es bisiesto</p> <p>    FinSi</p> <p>Fin</p>	 <pre> graph TD     Inicio([inicio]) --&gt; Decl[declaracion de el año]     Decl --&gt; Escribir[/escribir el año/]     Escribir --&gt; Leer[/leer el año/]     Leer --&gt; Div{el año es divisible por 400?}     Div -- Si --&gt; SiBisiesto[/si es bisiesto/]     Div -- no --&gt; NoBisiesto[/no es bisiesto/]     SiBisiesto --&gt; Fin([FIN])     NoBisiesto --&gt; Fin     </pre>



```

#include <stdio.h>

int main() {
    int anio;

    printf("Ingrese un año: ");
    scanf("%d", &anio);

    if (anio % 4 == 0) {
        if (anio % 100 == 0) {
            if (anio % 400 == 0) {
                printf("El año %d es bisiesto\n", anio);
            } else {
                printf("El año %d no es bisiesto\n", anio);
            }
        } else {
            printf("El año %d es bisiesto\n", anio);
        }
    } else {
        printf("El año %d no es bisiesto\n", anio);
    }

    return 0;
}

```

5) Crear una solución que valore el salario neto semanal de los trabajadores de una empresa de acuerdo a las siguientes normas:

- Hora semanales trabajadas <40 a una tasa.
- Horas extras (41 o más) a una tasa 50% superior a la ordinaria.
- Impuesto de 0%, si el salario bruto es menor o igual a 750 Balboas, 10%, si el salario bruto es mayor que 750 Balboas.

PSEUDOCODIGO	DIAGRAMA DE FLUJOS
--------------	--------------------

Inicio

Definir horas, tarifa  
Como Real  
Definir  
salarioBruto,  
salarioNeto,  
impuesto Como  
flotante

escribe Ingrese  
horas trabajadas en  
la semana:  
leer horas

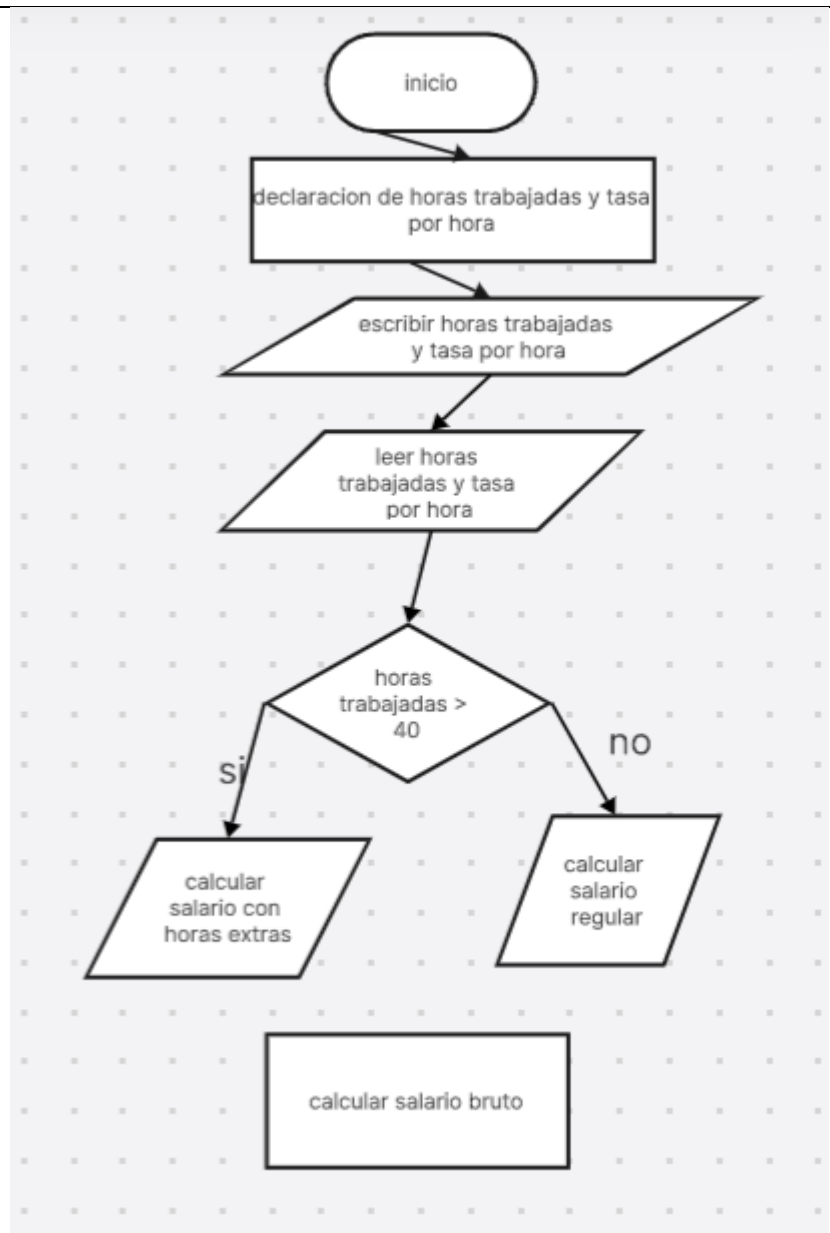
escribe Ingrese la  
tarifa por hora (B/.):  
leer tarifa

Si horas  $\leq 40$   
Entonces  
salarioBruto  $\leftarrow$   
horas \* tarifa  
Sino  
salarioBruto  $\leftarrow 40$   
\* tarifa + (horas - 40) \*  
(tarifa \* 1.5)  
Fin del Si

Si salarioBruto  $\leq$   
750 Entonces  
impuesto  $\leftarrow 0$   
Sino  
impuesto  $\leftarrow$   
salarioBruto \* 0.10  
Fin del Si

salarioNeto  $\leftarrow$   
salarioBruto -  
impuesto

escribe Horas  
trabajadas , horas



escribe Tarifa por  
hora: B/ , tarifa  
escribe Salario  
Bruto: B/ ,  
salarioBruto  
escribe Impuesto:  
B/, impuesto  
escribe Salario  
Neto: B/, salarioNeto  
Fin

```
#include <stdio.h>
#include <stdlib.h>

/* run this program using the console pauser or add your own getch, system("pause") or input loop */

int main(int argc, char *argv[]) {

    int horas;
    float tarifa, salarioBruto, salarioNeto, impuesto;

    printf("Ingrese horas trabajadas en la semana: ");
    scanf("%d", &horas);

    printf("Ingrese la tarifa por hora (B/.): ");
    scanf("%f", &tarifa);

    if (horas <= 40) {
        salarioBruto = horas * tarifa;
    } else {
        salarioBruto = 40 * tarifa + (horas - 40) * (tarifa * 1.5);
    }

    if (salarioBruto <= 750) {
        impuesto = 0;
    } else {
        impuesto = salarioBruto * 0.10;
    }

    salarioNeto = salarioBruto - impuesto;

    printf("\n--- Resumen de pago semanal ---\n");
    printf("Horas trabajadas: %d\n", horas);
    printf("Tarifa por hora: B/. %.2f\n", tarifa);
    printf("Salario Bruto: B/. %.2f\n", salarioBruto);
    printf("Impuesto: B/. %.2f\n", impuesto);
    printf("Salario Neto: B/. %.2f\n", salarioNeto);

    return 0;
}
```