

Universidad Tecnológica de Panamá
Facultad de Ingeniería en Sistemas Computacionales
Asignatura: Programación I
Examen Parcial1

Profesor: Napoleón Ibarra

Valor: 100 puntos

Estudiante: Felix Caballero

Cédula: 4-832-137

Fecha Inicio: 01/10/2025 → 2:30 PM

Fecha Entrega: 01/10/2025 → 4:55 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.

Criterios de Evaluación:

Criterios	Puntos (Mínimo=1, Máximo=!))	Porcentaje
Desarrollo	1-5	70 %
Sustentación	1-5	15 %
Responsabilidad	1-5	10 %

I PARTE. Desarrollo. Valor 50 puntos

- 1) 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.

```
2) import java.util.Scanner;
3)
4) public class Par1_1 {
5)     public static void main(String[] args) {
6)         Scanner sc = new Scanner(System.in);
7)         String resp;
8)
9)         do {
10)            System.out.println("=== INGRESO DE 3 FECHAS ===");
11)
12)            long[] fNum = new long[3];
13)            String[] fTxt = new String[3];
14)            boolean err = false;
15)
16)            for (int i = 0; i < 3; i++) {
17)                boolean ok = false;
```

```

19)         while (!ok && !err) {
20)             System.out.print("Ingrese fecha " + (i + 1));
21)             String in = sc.nextLine();
22)
23)             if (!in.contains(",")) {
24)                 System.out.println("Formato inválido. Use: dia, mes,
año");
25)                 continue;
26)             }
27)
28)             String[] p = in.split(",\\s*");
29)             if (p.length != 3) {
30)                 System.out.println("Formato inválido. Use: dia, mes,
año");
31)                 continue;
32)             }
33)
34)             try {
35)                 Double dVal = pedirNum("Validar día: " + p[0].trim(),
true, sc);
36)                 if (dVal == null) { err = true; break; }
37)                 int d = dVal.intValue();
38)                 if (d < 1 || d > 31) { System.out.println("Día fuera de
rango"); continue; }
39)
40)                 int m = mesNum(p[1].trim().toLowerCase());
41)                 if (m == -1) { System.out.println("Mes inválido");
continue; }
42)
43)                 Double yVal = pedirNum("Validar año: " + p[2].trim(),
false, sc);
44)                 if (yVal == null) { err = true; break; }
45)                 int y = yVal.intValue();
46)                 if (y < 1000 || y > 3000) { System.out.println("Año fuera
de rango"); continue; }
47)
48)                 fNum[i] = y * 10000L + m * 100L + d;
49)                 fTxt[i] = String.format("%.2f %.2f %.2f", (double)d,
(double)m, (double)y);
50)                 ok = true;
51)
52)             } catch (Exception e) {
53)                 System.out.println("Error: " + e.getMessage());
54)             }
55)         }
56)         if (err) break;
57)     }
58)
59)     if (!err) {

```

```

60)         System.out.println("\n=== FECHAS ===");
61)         for (String f : fTxt) System.out.println(f);
62)
63)         int min = 0, max = 0;
64)         for (int i = 1; i < 3; i++) {
65)             if (fNum[i] < fNum[min]) min = i;
66)             if (fNum[i] > fNum[max]) max = i;
67)         }
68)
69)         System.out.println("\n=== RESULTADOS ===");
70)         System.out.println("Más antigua: " + fTxt[min]);
71)         System.out.println("Más reciente: " + fTxt[max]);
72)     }
73)
74)     System.out.print("\n¿Desea ingresar otras 3 fechas? (s/n): ");
75)     resp = sc.nextLine();
76)
77)     } while (resp.equalsIgnoreCase("s"));
78)
79)     System.out.println("Fin del programa");
80)     sc.close();
81) }
82)
83) public static Double pedirNum(String msg, boolean pos, Scanner sc) {
84)     while (true) {
85)         System.out.print(msg + " ");
86)         try {
87)             String txt = sc.nextLine();
88)             double v = Double.parseDouble(txt);
89)
90)             if (pos && v < 0) {
91)                 System.out.println("No se permiten negativos");
92)             } else if (!Double.isFinite(v)) {
93)                 System.out.println("Número inválido");
94)             } else if (v < -1000 || v > 1000) {
95)                 System.out.println("Fuera de rango (-1000 a 1000)");
96)             } else {
97)                 return Math.round(v * 100.0) / 100.0;
98)             }
99)         } catch (NumberFormatException e) {
100)             System.out.println("Entrada inválida, solo números");
101)         } catch (Exception e) {
102)             System.out.println("Error inesperado: " + e.getMessage());
103)             return null;
104)         }
105)     }
106) }
107)
108) private static int mesNum(String m) {

```

```

109)         switch (m) {
110)             case "enero": return 1;
111)             case "febrero": return 2;
112)             case "marzo": return 3;
113)             case "abril": return 4;
114)             case "mayo": return 5;
115)             case "junio": return 6;
116)             case "julio": return 7;
117)             case "agosto": return 8;
118)             case "septiembre": return 9;
119)             case "octubre": return 10;
120)             case "noviembre": return 11;
121)             case "diciembre": return 12;
122)             default: return -1;
123)         }
124)     }
125) }
126)

```

- 2) 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.

```

3) import java.util.Scanner;
4)
5) public class Part1_2 {
6)     public static void main(String[] args) {
7)         Scanner sc = new Scanner(System.in);
8)         int n = pedirNum("Cuántas veces desea calcular la edad?", true,
9)             sc).intValue();
10)
11)         for (int i = 0; i < n; i++) {
12)             System.out.println("\n--- Cálculo " + (i + 1) + " ---");
13)
14)             int[] fn = pedirFecha("Fecha de nacimiento", sc);
15)             while (!validarFecha(fn[0], fn[1], fn[2])) {
16)                 System.out.println("Fecha de nacimiento inválida");
17)                 fn = pedirFecha("Fecha de nacimiento", sc);
18)             }
19)
20)             int[] fa = pedirFecha("Fecha actual", sc);
21)             while (!validarFecha(fa[0], fa[1], fa[2])) {
22)                 System.out.println("Fecha actual inválida");
23)                 fa = pedirFecha("Fecha actual", sc);
24)             }
25)             if (!validarOrden(fn, fa)) {

```

```

26)         System.out.println("La fecha actual debe ser posterior a la fecha
de nacimiento");
27)         i--;
28)         continue;
29)     }
30)
31)     calcularEdad(fn, fa);
32) }
33) sc.close();
34) }
35)
36) public static Double pedirNum(String msg, boolean pos, Scanner sc) {
37)     while (true) {
38)         System.out.print(msg + " ");
39)         try {
40)             String txt = sc.nextLine();
41)             double v = Double.parseDouble(txt);
42)
43)             if (pos && v < 0) {
44)                 System.out.println("No se permiten negativos");
45)             } else if (!Double.isFinite(v)) {
46)                 System.out.println("Número inválido");
47)             } else if (v < -10 || v > 3000) {
48)                 System.out.println("Fuera de rango.");
49)             } else {
50)                 return Math.round(v * 100.0) / 100.0;
51)             }
52)         } catch (NumberFormatException e) {
53)             System.out.println("Entrada inválida, solo números");
54)         } catch (Exception e) {
55)             System.out.println("Error: " + e.getMessage());
56)             return null;
57)         }
58)     }
59) }
60)
61) public static int[] pedirFecha(String msg, Scanner sc) {
62)     int[] f = new int[3];
63)     f[0] = pedirNum(msg + " - día (1-31): ", true, sc).intValue();
64)     f[1] = pedirNum(msg + " - mes (1-12): ", true, sc).intValue();
65)     f[2] = pedirNum(msg + " - año (4 dígitos): ", true, sc).intValue();
66)     return f;
67) }
68)
69) public static boolean validarFecha(int d, int m, int a) {
70)     if (m < 1 || m > 12) return false;
71)     int dm = diasMes(m, a);
72)     return d >= 1 && d <= dm;
73) }

```

```

74)
75)     public static int diasMes(int m, int a) {
76)         switch (m) {
77)             case 1: case 3: case 5: case 7: case 8: case 10: case 12: return 31;
78)             case 4: case 6: case 9: case 11: return 30;
79)             case 2: return (a % 4 == 0 && a % 100 != 0) || (a % 400 == 0) ? 29 :
28;
80)             default: return 0;
81)         }
82)     }
83)
84)     public static boolean validarOrden(int[] fn, int[] fa) {
85)         if (fa[2] > fn[2]) return true;
86)         if (fa[2] < fn[2]) return false;
87)         if (fa[1] > fn[1]) return true;
88)         if (fa[1] < fn[1]) return false;
89)         return fa[0] >= fn[0];
90)     }
91)
92)     public static void calcularEdad(int[] fn, int[] fa) {
93)         int a = fa[2] - fn[2];
94)         int m = fa[1] - fn[1];
95)         int d = fa[0] - fn[0];
96)
97)         if (d < 0) {
98)             m--;
99)             d += diasMes(fa[1] == 1 ? 12 : fa[1] - 1, fa[1] == 1 ? fa[2] - 1 :
fa[2]);
100)        }
101)
102)        if (m < 0) {
103)            a--;
104)            m += 12;
105)        }
106)
107)        if (a < 1) {
108)            System.out.println("La edad es: " + m + " meses y " + d + "
días");
109)        } else {
110)            double e = a + m / 12.0 + d / 365.0;
111)            System.out.printf("La edad es: %.2f años\n", e);
112)        }
113)    }
114) }
115)

```

- 3) Codificar una solución que determine si un año es bisiesto; esto se presenta cuando es múltiplo de 4, por ejemplo, 1984; sin 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.

```
4) import java.util.Scanner;
5)
6) public class Part1_3 {
7)     public static void main(String[] args) {
8)         Scanner sc = new Scanner(System.in);
9)         String cont = "s";
10)
11)         do {
12)             Double yD = pedirNum("Ingrese el año:", true, sc);
13)
14)             if (yD % 1 != 0) {
15)                 System.out.println("El año debe ser un número entero.");
16)                 continue;
17)             }
18)
19)             int y = yD.intValue();
20)             boolean bis = (y % 4 == 0) && (y % 100 != 0 || y % 400 == 0);
21)
22)             if (bis) {
23)                 System.out.println(y + " es un año bisiesto.");
24)             } else {
25)                 System.out.println(y + " no es un año bisiesto.");
26)             }
27)
28)             System.out.print("Desea verificar otro año?? (s/n): ");
29)             cont = sc.nextLine();
30)         } while (cont.equalsIgnoreCase("s"));
31)
32)         sc.close();
33)     }
34)
35)     public static Double pedirNum(String msg, boolean pos, Scanner sc) {
36)         while (true) {
37)             System.out.print(msg + " ");
38)             try {
39)                 String t = sc.nextLine();
40)                 double v = Double.parseDouble(t);
41)
42)                 if (pos && v < 0) {
43)                     System.out.println("No se permiten negativos");
44)                 } else if (!Double.isFinite(v)) {
45)                     System.out.println("Número inválido");
46)                 } else if (v < -10 || v > 3000) {
47)                     System.out.println("Fuera de rango.");
48)                 } else {
```

```

49)         return Math.round(v * 100.0) / 100.0;
50)     }
51) } catch (NumberFormatException e) {
52)     System.out.println("Entrada inválida, solo números");
53) } catch (Exception e) {
54)     System.out.println("Error! " + e.getMessage());
55)     return null;
56) }
57) }
58) }
59) }
60)

```

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

```

5) import java.util.Scanner;
6)
7) public class Part1_4 {
8)     public static void main(String[] args) {
9)         Scanner sc = new Scanner(System.in);
10)
11)         Double n = pedirNum("Cuántos trabajadores deseas procesar?", true, sc);
12)         if (n == null) return;
13)         int nt = n.intValue();
14)
15)         for (int i = 1; i <= nt; i++) {
16)             System.out.println("\n--- Trabajador " + i + " ---");
17)
18)             Double h = pedirNum("Horas trabajadas:", true, sc);
19)             if (h == null) continue;
20)
21)             Double t = pedirNum("Tasa ordinaria por hora:", true, sc);
22)             if (t == null) continue;
23)
24)             double sb;
25)             if (h <= 40) {
26)                 sb = h * t;
27)             } else {
28)                 double he = h - 40;
29)                 sb = (40 * t) + (he * t * 1.5);
30)             }
31)             sb = Math.round(sb * 100.0) / 100.0;
32)
33)             double imp = 0;

```



```

34)         if (sb > 750) {
35)             imp = sb * 0.10;
36)         }
37)         imp = Math.round(imp * 100.0) / 100.0;
38)
39)         double sn = sb - imp;
40)         sn = Math.round(sn * 100.0) / 100.0;
41)
42)         System.out.println("\nRESULTADOS:");
43)         System.out.println("Salario bruto: " + sb + " Balboas");
44)         System.out.println("Impuesto: " + imp + " Balboas");
45)         System.out.println("Salario neto: " + sn + " Balboas");
46)     }
47)
48)     sc.close();
49) }
50)
51) public static Double pedirNum(String msg, boolean pos, Scanner sc) {
52)     while (true) {
53)         System.out.print(msg + " ");
54)         try {
55)             String txt = sc.nextLine();
56)             double v = Double.parseDouble(txt);
57)
58)             if (pos && v < 0) {
59)                 System.out.println("No se permiten negativos");
60)             } else if (!Double.isFinite(v)) {
61)                 System.out.println("Número inválido");
62)             } else if (v < -10 || v > 3000) {
63)                 System.out.println("Fuera de rango.");
64)             } else {
65)                 return Math.round(v * 100.0) / 100.0;
66)             }
67)         } catch (NumberFormatException e) {
68)             System.out.println("Entrada inválida, solo números");
69)         } catch (Exception e) {
70)             System.out.println("Error: " + e.getMessage());
71)             return null;
72)         }
73)     }
74) }
75) }
76)

```

NOTA: Para el desarrollo de todos los problemas debe ser capaz de simularlo N veces. Usted (es) debe (n) tomar en cuenta 2 cifras significativas después del punto.

Figura 1. Plano de Oficina de la Universidad X

Requerimientos:

UNIVERSIDAD	EQUIPOS	IP	OBSERVACIÓN
X	2 IMPRESORA 1 SERVIDOR WEB JAVA (IT) 1 SERVIDAD DATOS (IT) 2 LAPTOPS (IT) 1 LAPTOPS (Dirección) 1 LAPTOPS (Subdirección) 1 LAPTOPS (Contabilidad) 1 LAPTOPS (Vicerrectoría) 1 PC (Recepción) 4 PC (Sala Profesores) 1 SWITCH 2960 DE 24 PUERTOS 1 ROUTER300N 2 EXTENSOR DE SEÑAL	IP=192.168.100.10 MR=255.255.255.0 PE=192.168.100.1 DNS1=8.8.8.8 DNS2=8.8.4.4	ISP 1: TYGO

UNIVERSIDAD	No. DE OFICINAS	DEPARTAMENTO	OBSERVACIÓN
X	No.1 No.2 No.3 No.4 No.5 No.6 No.7 No.8	Dirección Subdirección IT Vicerrectoria Académica Contabilidad Sala de Juntas Sala de Profesores Recepción	

BUENA SUERTE

