

# LENGUAJES DE PROGRAMACIÓN

Trabajo Práctico - Convocatoria ordinaria de 2025

## Instrucciones

- El alumno debe realizar por sí mismo el trabajo. No está permitido copiar los códigos solución de los ejercicios. El trabajo debe realizarse de manera individual. No debe realizarse en grupo. Se penalizará el plagio, así como cualquier uso compartido de las soluciones propuestas y de los códigos programados.
- El trabajo debe entregarse a través del curso virtual de la asignatura.
- La fecha límite de entrega es el día 16 de abril.
- El alumno debe entregar un fichero comprimido, en formato zip o tar, que contenga:
  - Un informe, en formato pdf, en el cual explique la solución a los ejercicios, incluyendo los listados documentados del código C++ desarrollado. Asimismo, en este documento se deben describir algunas de las pruebas realizadas para comprobar que los programas funcionan correctamente y deben mostrarse los resultados obtenidos en dichas ejecuciones de prueba.
  - Los ficheros del código fuente C++ solución a los ejercicios.

No deben entregarse ficheros ejecutables.

El nombre del fichero comprimido debe ser la concatenación de los apellidos y el nombre del alumno. Por ejemplo, GomezMartinLuisa.zip

## Criterios de evaluación

- Para que el trabajo pueda ser corregido, es imprescindible que el alumno entregue dentro del plazo establecido un fichero comprimido que contenga el informe en formato pdf y el código fuente C++ de los ejercicios que haya realizado.
- Si no entrega el informe, el trabajo se valorará con cero puntos.
- El trabajo se compone de 4 ejercicios, cada uno de los cuales se valorará sobre 2.5 puntos.
- No es obligatorio realizar todos los ejercicios. Para aprobar el trabajo es necesario y suficiente que la nota total obtenida en los ejercicios sea mayor o igual que 5.

- Si el código solución de un ejercicio tiene errores de compilación o no tiene la funcionalidad pedida, dicho ejercicio se valorará con cero puntos.

Se recomienda comprobar que el programa compila y ejecuta correctamente con el compilador online siguiente:

[https://www.onlinegdb.com/online\\_cplusplus\\_compiler](https://www.onlinegdb.com/online_cplusplus_compiler)

- Si el código solución de un ejercicio compila sin errores y tiene la funcionalidad pedida, la puntuación en dicho ejercicio será al menos de 2 puntos.
- Se valorará positivamente la adecuada documentación del código, así como la presentación y calidad de las explicaciones proporcionadas en el informe.

En el enunciado de los ejercicios se pide que describa en el informe algunas pruebas para comprobar que sus programas funcionan correctamente. Debe mostrar en el informe capturas de pantalla donde se muestren los resultados obtenidos en dichas ejecuciones de prueba.

## Ejercicio 1

El vector posición de una partícula puntual que se mueve en el espacio se define de la manera siguiente:

$$\vec{r} = a \cdot \cos(w \cdot t) \cdot \vec{u}_x + b \cdot \sin(w \cdot t) \cdot \vec{u}_y + c \cdot t \cdot \vec{u}_z$$

donde  $\vec{u}_x, \vec{u}_y, \vec{u}_z$  son los vectores unitarios de los ejes de coordenadas cartesianos X, Y, Z;  $t$  es una variable de tipo real que representa el tiempo;  $w, a, b, c$  son constantes conocidas de tipo real.

La longitud  $L$  de la trayectoria de la partícula entre los instantes  $t_A$  y  $t_B$ , satisfaciéndose  $t_A < t_B$ , puede calcularse de manera aproximada de la forma descrita a continuación.

1. Sea  $N$  un número entero mayor que cero.

2. Sean

$$t_i = t_A + i \cdot \frac{t_B - t_A}{N} \quad \text{para } i = 0, \dots, N$$

instantes de tiempo en el intervalo  $[t_A, t_B]$ .

3. Sea  $\vec{r}_i$  la posición de la partícula en el instante  $t_i$ .

4. Sea  $L_i$  la longitud del segmento recto que une la posición de la partícula en el instante  $t_i$  con la posición de la partícula en el instante  $t_{i+1}$ .

5. El valor aproximado de la longitud  $L$  se calcula de la forma siguiente:

$$L = \sum_{i=0}^{i=N-1} L_i$$

Escriba un programa en C++ que realice las acciones siguientes:

1. Declarar las constantes globales  $w, a, b, c$  asignándoles los valores siguientes:  $w = 10, a = 1.4, b = 2, c = 0.3$ .
2. Mostrar un mensaje en la consola solicitando al usuario que introduzca por consola el valor de  $N$  y los valores de los instantes  $t_A$  y  $t_B$ . Leer los valores introducidos por consola por el usuario.
3. Comprobar que se satisface  $0 \leq t_A < t_B$  y  $N > 0$ . En caso contrario, mostrar un mensaje en la consola indicándolo y terminar.

4. Calcular y mostrar en la consola la longitud aproximada de la trayectoria de la partícula entre los instantes  $t_A$  y  $t_B$ , calculada como se ha indicado anteriormente, y escribirla en la consola en formato científico con cinco dígitos decimales.
5. Terminar.

Muestre en el informe el resultado de ejecutar su programa para los casos de prueba siguientes:

<b>Caso de prueba 1:</b>	$N = 100,$	$t_A = 0.72,$	$t_B = 1.48$
<b>Caso de prueba 2:</b>	$N = 1000,$	$t_A = 0.72,$	$t_B = 1.48$
<b>Caso de prueba 3:</b>	$N = 10000,$	$t_A = 0.72,$	$t_B = 1.48$
<b>Caso de prueba Error 1:</b>	$N = -1,$	$t_A = 0,$	$t_B = 1$
<b>Caso de prueba Error 2:</b>	$N = 10,$	$t_A = -1,$	$t_B = 1$
<b>Caso de prueba Error 3:</b>	$N = 10,$	$t_A = 2,$	$t_B = 1$

## Ejercicio 2

Se desea calcular el importe de los intereses que deben abonarse en una cuenta bancaria cuyo funcionamiento es el siguiente: al finalizar cada año se abona en concepto de intereses el 2% del saldo medio existente en la cuenta durante dicho año.

En un fichero de texto están registrados los movimientos en la cuenta durante el año. El formato es el siguiente. En cada fila del fichero está registrado un movimiento. En la primera columna del fichero se escribe el día en que se ha producido el movimiento, en formato dd/mm/aaaa, en la segunda columna se escribe la hora a la cual se ha producido el movimiento, en formato HH:MM:SS, y en la tercera columna se indica el saldo existente en la cuenta una vez realizado el movimiento. Una de las filas del fichero indica el saldo existente a comienzo del año, esto es, el día 1 de enero a las 00:00:00 horas. Veamos un ejemplo, que corresponde a los movimientos de una cuenta durante el año 2024. Como puede observarse, los movimientos no se encuentran ordenados.

25/03/2024	12:14:02	7500.00
12/01/2024	13:11:01	9400.00
01/01/2024	00:00:00	12400.00
25/03/2024	22:59:02	7450.00
14/10/2024	07:11:03	10300.00

En el ejemplo anterior, el saldo al comienzo del año era de 12400 Euros. El día 12 de enero se realizó un movimiento, pasando el saldo de la cuenta a ser de 9400 Euros. El día 25 de marzo se realizaron dos movimientos: uno a las 12:14:02 y otro a las 22:59:02. El día 14 de octubre se realizó otro movimiento en la cuenta, tras el cual el saldo pasó a ser de 10300 Euros, manteniéndose la cuenta con ese saldo hasta finalizar el año 2024.

El saldo medio del año se calcula sumando el saldo en la cuenta al finalizar cada uno de los días del año y dividiendo por el número total de días del año (365 en año no bisiesto y 366 en año bisiesto). Obsérvese que se supone que el saldo de la cuenta cada día es igual al saldo existente en la cuenta al finalizar el día, es decir, a las 23:59:59 horas. En el ejemplo anterior, el día 25 de marzo se producen dos movimientos. Dado que al finalizar el día 25 de marzo el saldo de la cuenta era de 7450 Euros, se supone que ese fue el saldo de ese día.

En el ejemplo anterior, dado que el año 2024 fue bisiesto (febrero tuvo 29 días), el saldo promedio es:

$$S = \frac{12400 \cdot 11 + 9400 \cdot 73 + 7450 \cdot 203 + 10300 \cdot 79}{366} = 8602.87 \text{ Euros}$$

Deberán abonarse  $0.02 \cdot 8602.87 = 172.06$  Euros en concepto de intereses.

Escriba un programa en C++ que lea un fichero de texto llamado *movimientos.txt* en el cual están escritos los movimientos de una cuenta a lo largo de un año, y escriba en la consola el saldo medio anual y el importe que debe abonarse de intereses. Estas cantidades deben escribirse en formato fijo, con dos dígitos decimales.

Al escribir el programa puede suponer que el fichero *movimientos.txt* no contiene errores de formato. Puede suponer también que no contendrá varios movimientos realizados exactamente el mismo día y a la misma hora, minuto y segundo.

Muestre en el informe el resultado obtenido al ejecutar su programa cuando el contenido del fichero *movimientos.txt* es el siguiente:

25/03/2024	12:14:02	7500.00
12/01/2024	13:11:01	9400.00
01/01/2024	00:00:00	12400.12
25/03/2024	22:59:02	7450.00
14/10/2024	07:11:03	10300.14
19/03/2024	20:20:13	7537.92
25/03/2024	02:59:02	17450.00
25/03/2024	03:59:02	27450.10
25/03/2024	04:59:02	37450.45
20/05/2024	15:11:12	11034.65
12/07/2024	11:21:13	7648.27
11/11/2024	10:00:03	5637.12
29/12/2024	11:21:08	7986.36
01/01/2024	10:10:00	6400.26
08/09/2024	17:12:06	10761.89
12/05/2024	16:16:20	11987.42
13/05/2024	16:27:56	945.32
20/05/2024	11:11:12	12034.65
12/01/2024	09:11:27	9000.00
12/07/2024	13:25:23	7000.27
29/12/2024	16:22:03	8986.36

## Ejercicio 3

Consideremos el siguiente sistema de dos ecuaciones diferenciales ordinarias

$$\begin{aligned}\frac{dx}{dt} &= y \\ \frac{dy}{dt} &= \mu \cdot (1 - x^2) \cdot y - x\end{aligned}$$

donde  $t$  representa el tiempo y  $\mu$  es un parámetro real de valor conocido. En el instante inicial  $t_0$ , las variables  $x$  e  $y$  tienen valores conocidos:  $x(t_0) = x_0$ ,  $y(t_0) = y_0$ .

Se desea calcular la evolución de las variables  $x$  e  $y$  entre los instantes  $t_0$  y  $T$ , con  $t_0 < T$ . Para ello, se aplica el método de integración de Euler implícito, con un tamaño del paso de integración igual a  $h$ . Se define el entero  $N$  de la forma siguiente:

$$N = \left\lceil \frac{T - t_0}{h} \right\rceil$$

donde  $\lceil \cdot \rceil$  representa la *función techo*, la cual, aplicada a un número real, devuelve el menor número entero mayor o igual que el real.

El procedimiento iterativo de cálculo es el siguiente:

$$\begin{aligned}\frac{x_n - x_{n-1}}{h} &= y_n \\ \frac{y_n - y_{n-1}}{h} &= \mu \cdot (1 - x_n^2) \cdot y_n - x_n\end{aligned}$$

para  $n = 1, \dots, N$ , donde  $x_n$  e  $y_n$  representan respectivamente el valor de las variables  $x$  e  $y$  en el instante  $t_n = t_0 + n \cdot h$ .

Obsérvese que en el paso  $n$  son conocidos  $x_{n-1}$  e  $y_{n-1}$ , y deben calcularse  $x_n$  e  $y_n$  resolviendo el sistema de dos ecuaciones con dos incógnitas siguiente (se han señalado en color azul las incógnitas a calcular):

$$x_n - h \cdot y_n = x_{n-1} \quad (1.1)$$

$$h \cdot x_n + (1 - h \cdot \mu) \cdot y_n + h \cdot \mu \cdot x_n^2 \cdot y_n = y_{n-1} \quad (1.2)$$

Despejando  $x_n$  de la Ec. (1.1) y sustituyendo en la Ec. (1.2) obtenemos:

$$a \cdot y_n^3 + b \cdot y_n^2 + c \cdot y_n + d = 0 \quad (1.3)$$

donde los coeficientes  $a$ ,  $b$ ,  $c$ ,  $d$  se definen de la forma siguiente:

$$a = \mu \cdot h^3 \quad (1.4)$$

$$b = 2 \cdot \mu \cdot h^2 \cdot x_{n-1} \quad (1.5)$$

$$c = 1 + h^2 + h \cdot \mu \cdot (x_{n-1}^2 - 1) \quad (1.6)$$

$$d = h \cdot x_{n-1} - y_{n-1} \quad (1.7)$$

El cálculo de  $y_n$  de la Ec (1.3) puede hacerse aplicado el método de Newton:

$$y_{n,k+1} = y_{n,k} - \frac{a \cdot y_{n,k}^3 + b \cdot y_{n,k}^2 + c \cdot y_{n,k} + d}{3 \cdot a \cdot y_{n,k}^2 + 2 \cdot b \cdot y_{n,k} + c} \quad (1.8)$$

siendo  $k = 0, 1, \dots$  el índice de iteración del método de Newton. Como valor inicial de la iteración, escoja:  $y_{n,0} = y_{n-1}$ . La condición de finalización de la iteración del método de Newton es que el número de iteraciones alcance el valor  $k = 100$  o que se satisfaga

$$|a \cdot y_{n,k}^3 + b \cdot y_{n,k}^2 + c \cdot y_{n,k} + d| < TOL$$

Escriba un programa en C++ que resuelva numéricamente el sistema de dos ecuaciones diferenciales, empleando el procedimiento descrito. El programa debe declarar las siguientes constantes globales:  $t_0 = 0$ ,  $T = 20$ ,  $h = 10^{-6}$ ,  $\mu = 1.5$ ,  $TOL = 10^{-6}$ . El usuario debe introducir por consola las condiciones iniciales:  $x_0$ ,  $y_0$ .

El programa debe escribir en un fichero de texto llamado *resSimulacion.txt* los valores de las variables  $t$ ,  $x$  e  $y$  en los instantes siguientes:  $t_0$ ,  $t_0 + H$ ,  $t_0 + 2 \cdot H$ ,  $\dots$ , con  $H = 10000 \cdot h$ .

El formato del fichero debe ser el siguiente: en la primera columna se escribe el valor del tiempo, en la segunda columna el valor de  $x$  en ese instante y en la tercera columna el valor de  $y$  en ese instante. Todo ello en formato fijo, con 4 dígitos decimales. Las columnas deben estar separadas por un espacio en blanco.

Muestre en el informe las 10 primeras filas y las 10 últimas filas del fichero generado por su programa cuando es ejecutado para las condiciones iniciales siguientes:  $x_0 = 2$ ,  $y_0 = 0$ .

## Ejercicio 4

En un fichero de texto llamado *pisos.txt* se encuentra almacenada información acerca de pisos ofertados en alquiler en una ciudad. La información acerca de un piso está descrita mediante una secuencia de expresiones del tipo IDENT=N separadas entre sí por uno o más espacios en blanco o saltos de línea, donde IDENT es uno de los identificadores mostrados en la tabla siguiente y N es un número entero. La información correspondiente a un piso finaliza con un punto y coma (;) precedido y sucedido de uno o varios espacios en blanco y/o saltos de línea.

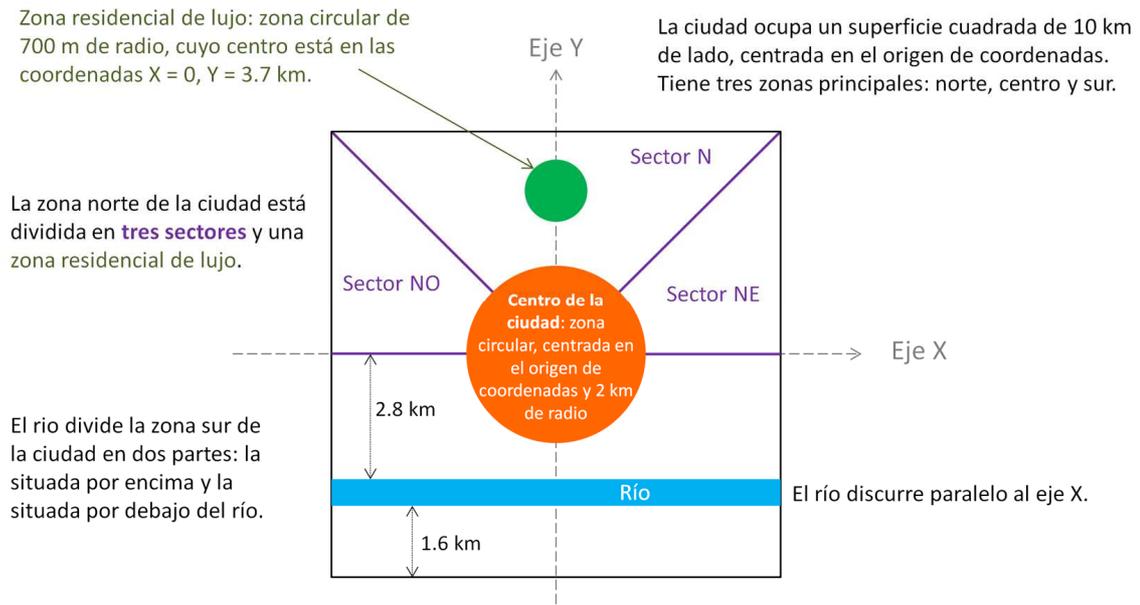
IDENT	Significado
PLAN	Número de la planta donde está el piso.
EDIF	Número de plantas del edificio.
DORM	Número de dormitorios que tiene el piso.
BAGN	Número de baños.
SUPT	Superficie total del piso (m <sup>2</sup> ).
SUPU	Superficie útil del piso (m <sup>2</sup> ).
POSX	Coordenada X (m) de la ubicación del piso en la ciudad.
POSY	Coordenada Y (m) de la ubicación del piso en la ciudad.
PREC	Importe del alquiler mensual.

Por ejemplo, las dos siguientes líneas del fichero

```
PLAN=2 PREC=1000 POSX=2300 POSY=300 EDIF=14
DORM=3 BAGN=1 SUPT=130 SUPU=70 ;
```

describen un piso ubicado en una segunda planta de un edificio de catorce plantas, que tiene tres dormitorios, un baño, una superficie total de 130 m<sup>2</sup> y una superficie útil de 70 m<sup>2</sup>, se encuentra en la coordenadas (X=2300 m, Y=300 m) del plano de la ciudad, y tiene un precio mensual de alquiler de 1000 Euros. Obsérvese que las expresiones pueden estar escritas en cualquier orden.

En la figura de la página siguiente están representadas las distintas zonas de la ciudad, desde el punto de vista del mercado del alquiler. La ciudad ocupa una superficie cuadrada de 10 km de lado. Para describir la ubicación de los pisos en la ciudad, se emplea un sistema de coordenadas cartesiano, con los ejes paralelos a los lados del cuadrado y con el origen de coordenadas en el centro del cuadrado.



La zona denominada *Centro de la Ciudad* es un círculo de radio 2 km cuyo centro coincide con el centro de la ciudad. La zona de la ciudad situada por encima del eje X, excluyendo el *Centro de la Ciudad*, se denomina *Zona Norte* y está dividida en tres sectores denominados *Sector NO* (noroeste), *Sector N* (norte) y *Sector NE* (noreste), y una *Zona Residencial de Lujo*, que ocupa una zona circular de 700 m de radio centrada en  $(X=0, Y=3.7$  km). Los dos segmentos de recta diagonales que delimitan los sectores intersectan en el origen de coordenadas. La zona de la ciudad situada por debajo del eje X, excluyendo el *Centro de la Ciudad*, se denomina *Zona Sur* y se compone de dos partes: la situada por encima del río y la situada por debajo del río.

El criterio para determinar a qué zona corresponden los pisos que se encuentran sobre la frontera entre zonas es el siguiente. Si el piso se encuentra sobre la frontera de la zona centro o residencial, pertenece a la zona centro o residencial respectivamente. Si no se satisface lo anterior y el piso se encuentra en la frontera de los Sectores NO o NE, pertenece a los Sectores NO o NE respectivamente.

Escriba un programa en C++ que solicite al usuario que introduzca por consola la descripción del tipo de piso que busca, especificando para ello todo lo siguiente:

- Número mínimo de dormitorios y número mínimo de baños.
- Número mínimo de metros cuadrados útiles.
- Importe máximo que está dispuesto a pagar mensualmente por el alquiler.

- Zona de la ciudad donde desea que se encuentre el piso: (0) Centro; (1) Sector NO; (2) Sector N; (3) Sector NE; (4) Residencial de Lujo; (5) Sur encima del río; o (6) Sur debajo del río.

El programa debe leer el fichero *pisos.txt* y mostrar en la consola la descripción de los pisos que cumplen simultáneamente todas esas condiciones, ordenados de menor a mayor importe mensual del alquiler.

Puede suponer que el fichero *pisos.txt* no tiene errores de formato y que las descripciones de los pisos están completas (en cada piso se asigna valor a todos los identificadores). A continuación se muestra el contenido del fichero *pisos.txt* (continúa en la página siguiente).

```

PLAN=2 PREC=1300 POSX=2000 POSY=0 EDIF=14
DORM=3 BAGN=1 SUPT=130 SUPU=70 ;
PLAN=4 PREC=1100 POSX=-2000 POSY=0 EDIF=6
DORM=3 BAGN=1 SUPT=120 SUPU=90
;
PLAN=2 EDIF=14 POSY=-2000 POSX=0 PREC=1200 SUPT=90 SUPU=65
  DORM=2 BAGN=1 ;
PLAN=8 EDIF=10 POSY=500 POSX=500 PREC=2000 SUPT=180 SUPU=140
  DORM=4 BAGN=2 ;
PLAN=1 PREC=1300 POSX=-550 POSY=250 EDIF=4
DORM=3 BAGN=1 SUPT=120 SUPU=85 ;
PLAN=3 PREC=2500 POSX=1000 POSY=-1000 EDIF=4 DORM=4 BAGN=2 SUPT=160 SUPU=120 ;
PLAN=1 EDIF=3 POSY=-300 POSX=-350 PREC=800 SUPT=60 SUPU=35
BAGN=1 DORM=0 ;
PLAN=1 EDIF=3 POSY=50 POSX=-50 PREC=2800 SUPT=220 SUPU=160 DORM=4 BAGN=2 ;
POSY=0 POSX=-3000 PREC=1800
PLAN=1 EDIF=3 SUPT=220 SUPU=160 DORM=4 BAGN=2 ;
POSY=2500 POSX=-2500 PREC=1400
PLAN=2 EDIF=4 SUPT=160 SUPU=110 DORM=3 BAGN=2 ;
POSY=3000 POSX=-3000 PREC=1400
PLAN=3 EDIF=4 SUPT=140 SUPU=105 DORM=3 BAGN=2 ;
POSY=400 POSX=-4000 PREC=1550
PLAN=6 EDIF=8 SUPT=140 SUPU=120 DORM=4 BAGN=2 ;
POSY=3000 POSX=-3000 PREC=800
PLAN=6 EDIF=8 SUPT=70 SUPU=60 DORM=1 BAGN=1 ;
PLAN=1 EDIF=3 POSX=-500 POSY=2550 PREC=600 SUPT=50 SUPU=30
BAGN=1 DORM=0 ;
PLAN=2 EDIF=4 POSX=500 POSY=2000 PREC=700 SUPT=80 SUPU=60
BAGN=1 DORM=2 ;
POSX=0 POSY=4500 PREC=1200

```

LENGUAJES DE PROGRAMACIÓN

PLAN=3 EDIF=10 SUPT=160 SUPU=100 DORM=3 BAGN=2 ;  
PO SX=0 POSY=3750 PREC=2200  
PLAN=1 EDIF=3 SUPT=160 SUPU=100 DORM=3 BAGN=2 ;  
PO SX=0 POSY=3000 PREC=1800  
PLAN=2 EDIF=4 SUPT=180 SUPU=130 DORM=4 BAGN=2 ;  
PO SX=500 POSY=3700 PREC=3000  
PLAN=1 EDIF=1 SUPT=400 SUPU=300 DORM=6 BAGN=3 ;  
PLAN=1 EDIF=3 POSY=0 POSX=2050 PREC=1200 SUPT=120 SUPU=90 DORM=3 BAGN=2 ;  
PLAN=2 EDIF=5 POSY=2350 POSX=2900 PREC=1000 SUPT=100 SUPU=80 DORM=3 BAGN=2 ;  
PLAN=4 EDIF=10 POSY=700 POSX=4000 PREC=900 SUPT=100 SUPU=80 DORM=3 BAGN=1 ;  
PLAN=6 EDIF=12 POSY=3000 POSX=3000 PREC=800 SUPT=60 SUPU=50 DORM=1 BAGN=1 ;  
PLAN=3 EDIF=4 POSY=3550 POSX=4900 PREC=600 SUPT=50 SUPU=35 DORM=0 BAGN=1 ;  
PLAN=6 EDIF=6 POSX=-2500 POSY=-2550 PREC=600 SUPT=50 SUPU=30  
BAGN=1 DORM=0 ;  
PLAN=4 EDIF=6 POSX=-500 POSY=-2550 PREC=650 SUPT=50 SUPU=30  
BAGN=1 DORM=0 ;  
PLAN=2 EDIF=6 POSX=600 POSY=-2550 PREC=680 SUPT=50 SUPU=35  
BAGN=1 DORM=0 ;  
PLAN=2 EDIF=4 POSX=-500 POSY=-2600 PREC=690 SUPT=55 SUPU=30  
BAGN=1 DORM=0 ;  
PLAN=1 EDIF=4 POSX=0 POSY=-2550 PREC=650 SUPT=55 SUPU=37  
BAGN=1 DORM=0 ;  
PLAN=4 EDIF=8 POSX=-2500 POSY=-2800 PREC=690 SUPT=60 SUPU=40  
BAGN=1 DORM=0 ;  
PLAN=6 EDIF=6 POSX=2500 POSY=-2500 PREC=1000 SUPT=100 SUPU=80  
BAGN=1 DORM=3 ;  
PLAN=6 EDIF=6 POSX=500 POSY=-2500 PREC=1200 SUPT=120 SUPU=90  
BAGN=2 DORM=3 ;  
PLAN=6 EDIF=6 POSX=-2500 POSY=-3550 PREC=450 SUPT=50 SUPU=30  
BAGN=1 DORM=0 ;  
PLAN=4 EDIF=6 POSX=-500 POSY=-3750 PREC=550 SUPT=50 SUPU=30  
BAGN=1 DORM=0 ;  
PLAN=2 EDIF=6 POSX=600 POSY=-4550 PREC=580 SUPT=50 SUPU=35  
BAGN=1 DORM=1 ;  
PLAN=2 EDIF=4 POSX=-500 POSY=-3600 PREC=490 SUPT=55 SUPU=30  
BAGN=1 DORM=0 ;  
PLAN=1 EDIF=4 POSX=0 POSY=-3550 PREC=550 SUPT=55 SUPU=37  
BAGN=1 DORM=0 ;  
PLAN=4 EDIF=8 POSX=-2500 POSY=-3800 PREC=600 SUPT=60 SUPU=40  
BAGN=1 DORM=1 ;  
PLAN=3 EDIF=6 POSX=2500 POSY=-4600 PREC=900 SUPT=100 SUPU=80  
BAGN=2 DORM=3 ;  
PLAN=6 EDIF=6 POSX=500 POSY=-4900 PREC=1000 SUPT=120 SUPU=90  
BAGN=2 DORM=4 ;

Muestre en el informe el resultado de ejecutar su programa para los tres casos de prueba indicados a continuación.

### **Caso de prueba 1:**

Numero minimo de dormitorios: 2  
Numero minimo de bagnos: 1  
Numero minimo de m2 utiles: 80  
Importe maximo alquiler: 1000

Zonas

- (0) Centro
- (1) Sector NO
- (2) Sector N
- (3) Sector NE
- (4) Residencial Lujo
- (5) Sur encima rio
- (6) Sur debajo rio

Seleccione zona: 6

### **Caso de prueba 2:**

Numero minimo de dormitorios: 3  
Numero minimo de bagnos: 1  
Numero minimo de m2 utiles: 80  
Importe maximo alquiler: 1600

Zonas

- (0) Centro
- (1) Sector NO
- (2) Sector N
- (3) Sector NE
- (4) Residencial Lujo
- (5) Sur encima rio
- (6) Sur debajo rio

Seleccione zona: 0

### **Caso de prueba 3:**

Numero minimo de dormitorios: 2  
Numero minimo de bagnos: 1  
Numero minimo de m2 utiles: 90  
Importe maximo alquiler: 1500

Zonas

- (0) Centro
- (1) Sector NO
- (2) Sector N
- (3) Sector NE
- (4) Residencial Lujo
- (5) Sur encima rio
- (6) Sur debajo rio

Seleccione zona: 2