Examen noviembre 2007, IND.

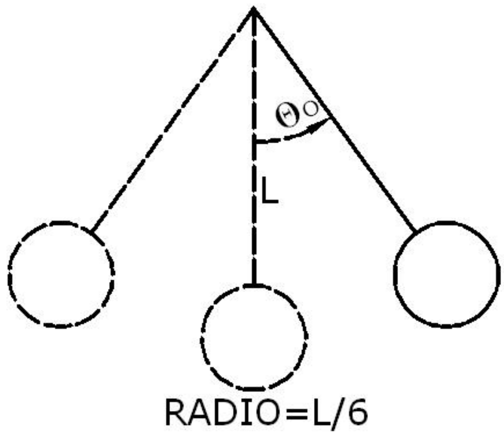	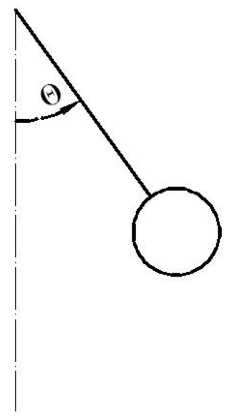
Figura 3. Representación gráfica de un péndulo simple. El ángulo inicial es = θ_0 y la varilla de longitud L está anclada en su extremo superior.	Figura 4. Gráfico correspondiente a la posición de ángulo θ del péndulo. PINS es el extremo superior de la varilla.

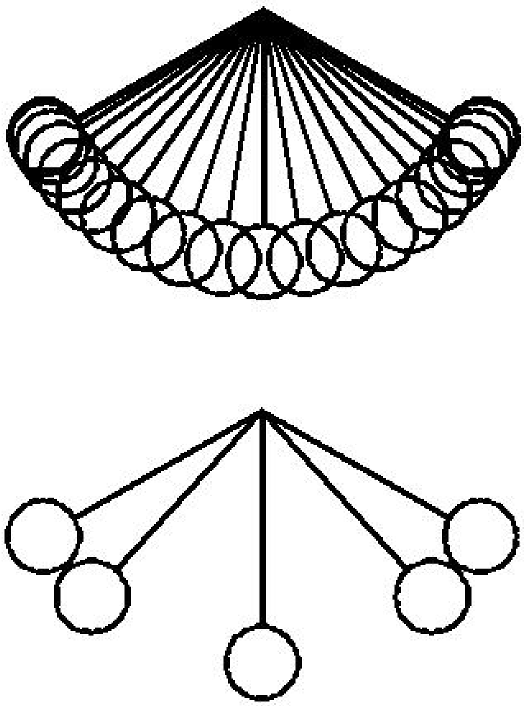

Figura 5. La ejecución de arriba ha surgido de esta invocación: (GRAFICO (LIST 0 0) (/ PI 3) 41 20))
La ejecución de debajo, de ejecutar el comando C:PENDULO contestando:
PUNTO DE BASE PENDULO **0,0**
VALOR INICIAL THETA0, EN SEXAGESIMALES SI VA DE TECLADO **60**
NÚMERO DE POSICIONES DEL PÉNDULO EN UN CICLO **41**
LONGITUD DE LA VARILLA **20**
INTERVALO ENTRE CADA REPRESENTACIÓN GRÁFICA **5**

El resultado es que se dibuja 1 DE CADA 5 posiciones calculadas

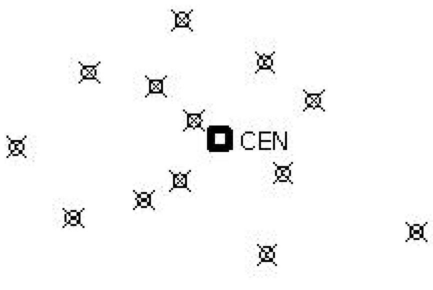

Figura 6. El CENTROIDE de una nube de puntos es su centro de masas. Se obtiene mediante el cálculo de las medias de las coordenadas x e y:

$$Xcen = \frac{\sum_{i=1}^{n} Xi}{n} \quad ; \quad Ycen = \frac{\sum_{i=1}^{n} Yi}{n}$$

Estructuras Cíclicas. Dibujo de curvas.

El movimiento oscilatorio de un péndulo simple (figura 3) queda definido numéricamente por la expresión:

$$\theta = \theta_0 \,.\, sen\ (a + \pi/2), \qquad \underline{ec1}$$

donde a toma valores en el intervalo [0, m], siendo m un valor real cualquiera.

1. Escribir una función VLISP **POSICION (A THETA0 / . . .)** que devuelva el valor del ángulo **θ** que corresponde al valor del parámetro a según la expresión ***ec1***. (0,5p)

2. Escribir una función VLISP **POSICIONES (M THETA0 DELTA / . . .)** que devuelva la lista **(θ_0 θ_1 . . .)** con los valores del ángulo **θ** que corresponden a los valores de **a** en el intervalo **[0, m]** saltando en incrementos de valor **delta**. (0,5p)

3. Escribir una función VLISP **DIBUJA (PINS THETA L / . . .)** que dibuje el péndulo en la posición definida por el ángulo theta. Ver figura 4. (0,5p)

4. Escribir una función VLISP **GRAFICO (PINS THETA0 N L / .)** que dibuje las posiciones del péndulo bajo estas especificaciones: (0,5p)

- El intervalo de existencia de a es **[0,2π]**.
- ***n*** es el número de posiciones que se desea obtener, incluidas las de los extremos del intervalo. El incremento angular delta debe proponerse dividiendo el intervalo **[0,2π]** en partes iguales.
- **L** es la longitud de la varilla del péndulo y **L/6** es el radio con que se representa la masa que oscila al final de la varilla. Ver de nuevo la figura 3.
- Usar FOREACH.

5. Reescribir la función anterior en forma de nuevo comando de Autocad de nombre **C:PENDULO (/ . . .)**.
En este apartado, esta función no dibujará las n posiciones del péndulo, sino que lo hará de S en S, siendo este parámetro S una variable entera que el comando también solicitará al usuario. Ver figura 5 (2p)

Estructuras Cíclicas. Búsqueda y ordenación.

6. Dada la lista **LP = (P_0 P_1 …)**, escribir una función VLISP **CENTROIDE (LP / . .)** que devuelva la lista CEN(Xcen Ycen) con las coordenadas de su centroide. Usar REPEAT. Ver figura 6. (0,5p)

7. Dada una lista **LP = (P_0 P_1 …)**, escribir una función VLISP **PRÓXIMO (LP / . .)** que devuelva el punto Pj de la lista LP más próximo al punto CEN. (1,3p)

8 Dada una lista **LP = (P_0 P_1 …)**, escribir una función VLISP **ORDENADIS (LP / . .)** que calcule la lista de distancias de los puntos Pj de LP a CEN y la devuelva ordenada de menor a mayor. (2,2p)

UC UNIVERSIDAD DE CANTABRIA

Examen septiembre 2007, TEL.

BLOQUE 1. Funciones geométricas, etc.

1. Escribir una función VLISP **ANGULOS (P B1 B2 / . . .)** que devuelva la lista con los valores (ang1 ang2), siendo B1 y B2 las medidas de los lados del triángulo. Ver figura 1.
NOTA: no hace falta comprobar que OP < B1+B2; se supone que los datos con que se invoca esta función ya vienen cumpliendo esta condición. (1p)

BLOQUE 2. Invocación a funciones; funciones de control.

2. Escribir una función VLISP **BRAZO (P B0 B1 / . . .)** que dibuje el brazo articulado tal como se muestra en la figura 2. Para ello, la función primero comprobará si se cumple OP < B0+B1; si es así, dibujará el brazo y si no lo es, indicará el error y detendrá la ejecución del programa sin dibujar nada. (1p)

BLOQUE 3. Ciclos, Listas. Dibujo de curvas.

Se conocen las ecuaciones paramétricas del movimiento del extremo de un brazo articulado:

$$X = a_2 t^2 + a_1 t + a_0$$

$$Y = b_3 t^3 + b_2 t^2 + b_1 t + b_0$$

Donde el parámetro t representa el tiempo y debe tomar valores en el intervalo [0,TMAX].

Los símbolos TMAX, a_0, a_1, a_2, b_0, b_1, b_2 y b_3 son variables globales que ya tienen su valor asignado en el momento de ejecutar las funciones que se describen a continuación.

3.1. Escribir la función VLISP **POSICION (t_j / . . .)** que devuelva:

- NIL si t_j no pertenece al intervalo [0,TMAX].
- La lista P_j (X_j Y_j) correspondiente al instante tj en caso contrario. (1p)

3.2.. Escribir una función VLISP **TRAYECTORIA (t_i t_f / . . .)** que devuelva la lista LP = (P_i . . . P_f) con los puntos de la trayectoria recorrida por el brazo desde el instante ti hasta el tj, a intervalos de valor (t_f – t_i)/100.
NOTA: es posible que la lista contenga uno o varios valores NIL en sus extremos, debido a que el parámetro t no esté dentro del intervalo [0,TMAX]. (2p)

3.3. Escribir una función VLISP **DIBUJA_TRAYECTORIA (t_i t_f / . . .)** que dibuje el contenido de la lista LP como una lwpolilínea.
NOTA: hay que controlar la posibilidad de que LP tenga valores NIL en sus extremos, debido a que el parámetro t no esté dentro del intervalo [0,TMAX]. La función debe

UC
UNIVERSIDAD DE CANTABRIA

ignorar estos valores NIL, comenzando la curva en el primer valor no NIL y acabándola en el último valor no NIL. (2p)

3.4. Escribir una función VLISP **VELOCIDAD (t_i / . . .)** que devuelva la lista V (MOD_v $ORIENT_v$) con los valores del Módulo y la Orientación del vector velocidad para $t=t_i$.
NOTAS: V podría valer NIL si t_i no está dentro del intervalo [0,TMAX]. La orientación se entiende en los términos usuales estudiados en VLISP. (3p)

TIEMPO: 1 HORA Y 45 MINUTOS
Excepto las indicadas al principio del bloque 3, TODAS LAS VARIABLES HAN DE SER LOCALES.

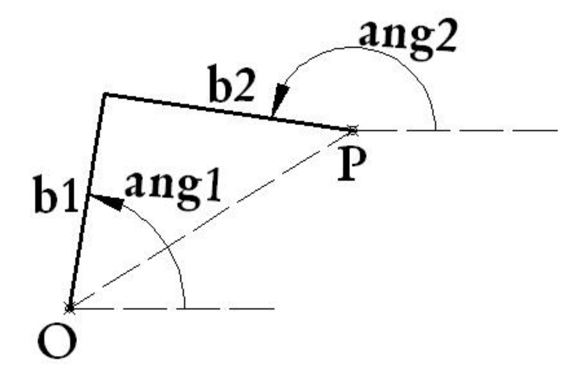

Figura 1. Los valores devueltos son las orientaciones en radianes de los segmentos orientados b1 y b2 con el sentido de las X positivas y con sentido antihorario. El segmento orientado b1 tiene su origen en O y el b2 tiene su origen en P. O es el origen de coordenadas

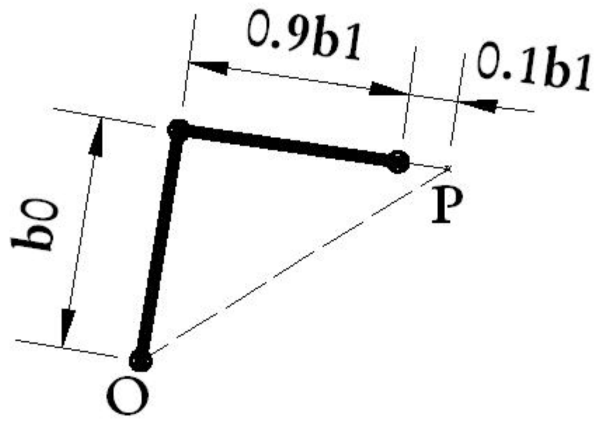

Figura 2. La representación del brazo articulado requiere sólo el dibujo de tres círculos en la CAPA1, dos líneas en la CAPA2 y una línea en la CAPA3. Los espesores adecuados ya están definidos en esas CAPAS, de modo que no hay que preocuparse por ese aspecto en la programación de la función VLISP propuesta.
Los círculos tienen de radio 1 ud.
La medida de los segmentos rectos es la que se acota en la propia figura.

Examen Febrero 2007. Ind. **Estructuras Cíclicas. Dibujo de curvas.**

3. *Las siguientes ecuaciones polares paramétricas corresponden a curvas clásicas de rodadura:*

- ***Círculo,*** $C(\rho,\theta):\begin{cases}\theta = u \\ \rho = k.\cos u\end{cases}$, *u € [0, π], k es una constante.*

- ***Quadrifolio,*** $Q(\rho,\theta):\begin{cases}\theta = u/2 \\ \rho = k.\cos u\end{cases}$, *u € [0, π], k es una constante.*

- ***Limaçon,*** $L(\rho,\theta):\begin{cases}\theta = 3.u \\ \rho = k.\cos u\end{cases}$, *u € [0, π], k es una constante.*

3.1. Escribir una función **VLISP PUNTOCR (u TIPO K / . . .)** que devuelva el punto de la curva PP(rho theta), teniendo en cuenta:

* Que el argumento u es un valor real perteneciente a [0, π]. Representa radianes.
* Que TIPO es un alfanumérico que puede valer “C”, “Q” o “L”, según se refiera a un Círculo a un Quadrifolio o a una Limaçon.
* Que K es un valor real.

Los valores de Rho y Theta se obtendrán de las expresiones indicadas más arriba. (1p)

3.2. Escribir una función VLISP **CURVARODADURA (TIPO K N / . . .)** que devuelva la lista de puntos (PP_0 PP_1 . . . PP_n) de la curva indicada por el parámetro TIPO correspondientes a N valores de u en el intervalo [0, π], incluidos estos dos. La función no dibuja nada. K es un valor real. (1p)

3.3. Escribir una función VLISP **POLARESACARTESIANAS (LPP / . . .)** que reciba una lista LPP = (PP_0 PP_1 . . . PP_n) de puntos PP(rho theta) expresados en polares y devuelva una lista LP = (P_0 P_1 . . . P_n) con esos mismos puntos P(X Y) expresados en cartesianas. (0.5)

3.4. Escribir una función VLISP **LONGITUDCURVA (LPP / . . .)** que reciba una lista LPP = (PP_0 PP_1 . . . PP_n) de puntos PP(rho theta) expresados en polares y devuelva su longitud como suma de la poligonal PP_0-PP_1 -. . .- PP_n. (0.5)

3. Una parábola $\mathscr{P}$ como la que se representa en la figura 4 tiene esta expresión paramétrica:

$P(x, y): \begin{cases} x = 2kt^2 \\ y = 2kt \end{cases}$, donde el parámetro t puede tomar valores en $[0,\infty]$. K es una constante que el usuario elije para que la parábola sea más abierta o más cerrada.

Escribir una función VLISP **PARABOLA (N K / . . .)** que devuelva la lista de puntos (P0 P1 . . . Pn) de la parábola $\mathscr{P}$ correspondientes a los valores de t enteros en el intervalo [0,N]. K es la constante de la parábola . La función no dibuja nada. (1p)

4. La pendiente de la recta normal en un punto P (X_p Y_p) de la parábola $\mathscr{P}$ vale m=-2t. Escribir una función VLISP **PUNTO_EN_NORMAL (P D / . . .)** que devuelva el punto PNOR (X_{pnor} Y_{pnor}) sujeto a estas condiciones:

- P es un punto de la parábola
- D es una distancia que propone el usuario
- PN dista D del punto P y el segmento Pn-P es normal a la parábola. Ver la figura 5. (1p)

5. Escribir una función VLISP **PARABOLA_EQUIDISTANTE (N K D / . . .)** que devuelva la lista de puntos ($PNOR_0$ $PNOR_1$ . . . $PNOR_n$) correspondientes a una parábola equidistante a la $\mathscr{P}$ a una distancia D. La función no dibuja nada. (1,5p)

En realidad hay dos parábolas equidistantes, a cada uno de los lados de $\mathscr{P}$. Da igual cuál de las dos se elija.

6. Escribir una programa VLISP **C:PARABOLAS ( )** que solicite interactivamente los valores **N** y **K** de la parábola $\mathscr{P}$ y el valor **D** de su parábola equidistante y dibuje ambas. Ver la figura 6. (1,5p)

Estructuras cíclicas. Ordenación y manejo de listas.

7. Una lista LTR = (P_0 P_1 P_2) representa un triángulo en el espacio 3D. Cada vértice de esa lista es otra lista de punto P_j = (X_j Y_j Z_j).

Escribir una función VLISP **ORDENA_TRIANGULO (LTR / . . .)** que devuelva la lista LTR ordenada por los puntos Pj de mayor a menor coordenada Z. Ver la figura 7. (0,5p)

8. Sea una lista CARAS = (LTR_0 LTR_1 . . . LTR_M), donde cada LTR es una lista (P_0 P_1 P_2) como la que se indica en el apartado anterior. Se considera que una cara LTR_i es más alta que otra LTR_j cuando se cumplen los criterios que se indican en la figura 8.

Escribir una función VLISP **ORDENA_CARAS (CARAS / . . .)** que devuelva la lista CARAS ordenada por sus caras en sentido de altura decreciente (de mayor a menor altura). (2,5p)

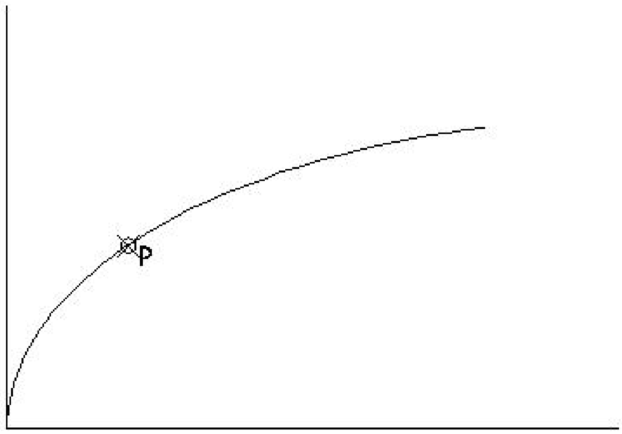

Figura 4. La parábola $\mathscr{P}$ está constituida por puntos cuya expresión paramétrica se propone en el apartado 4.

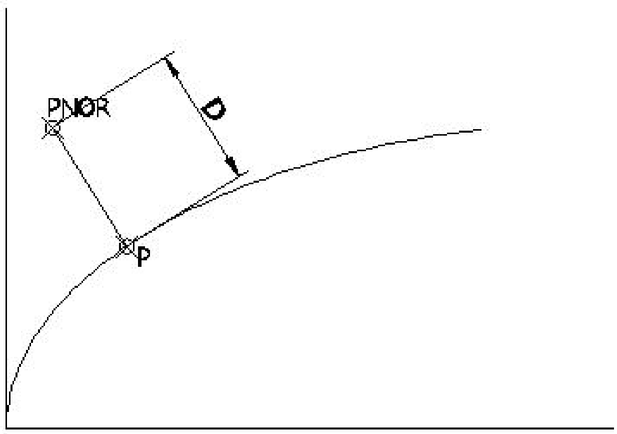

Figura 5. A partir del punto P se puede calcular la orientación de la normal a la parábola en ese punto y así definir PNOR.

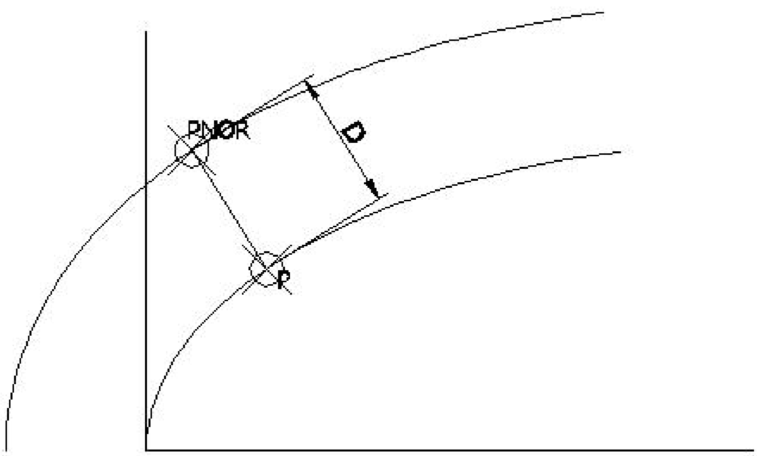

Figura 6. Al variar t en [0,n], P recorre la parábola $\mathscr{P}$ y PNOR su equidistante.

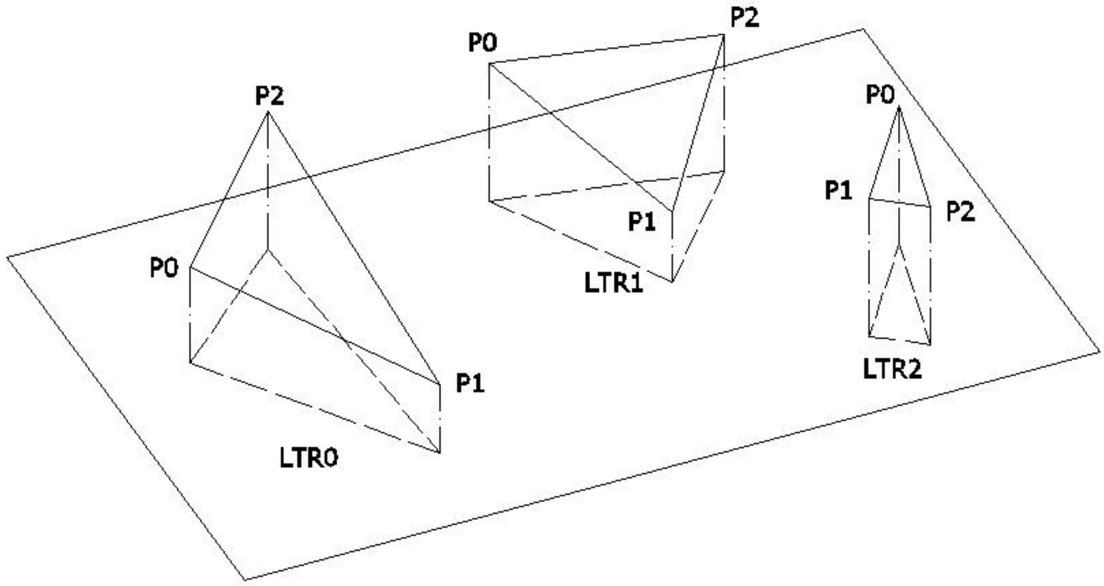

Figura 7.En el triángulo LTR_0, P_2 tiene mayor z que P_0 y éste que P_1, de modo que la secuencia ordenada de sus vértices es (P_2 P_0 P_1); en LTR1 la secuencia ordenada puede ser (P_0 P_2 P_1) o (P_2 P_0 P_1), ya que P0 y P2 tienen la misma z. En LTR2, cualquier ordenación es válida porque los tres tienen la misma z.

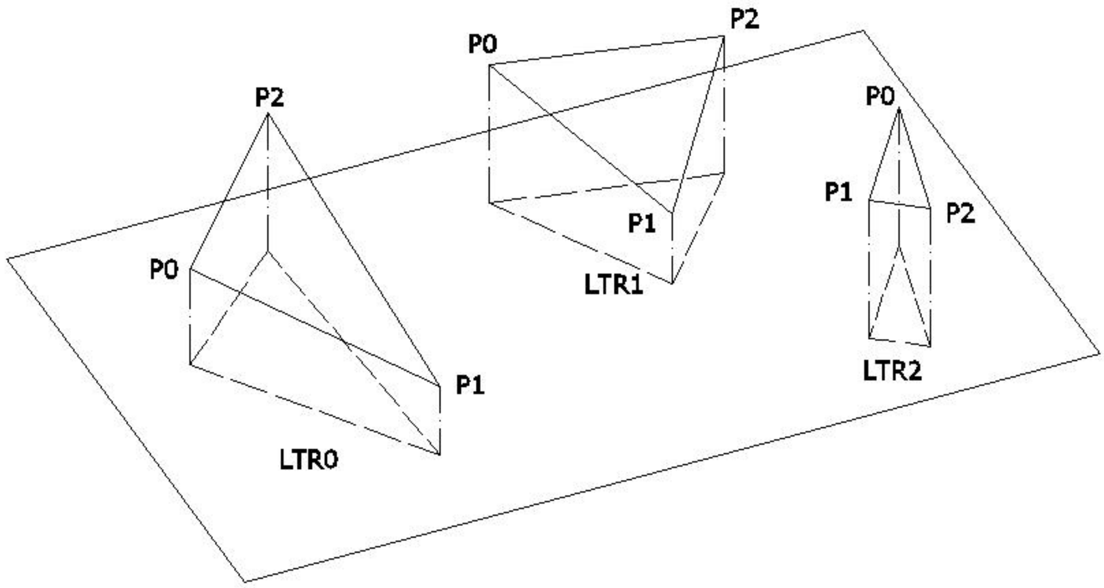

Figura 8. Para ordenar las caras, lo primero que se hace es ordenar cada LTRi según el criterio del apartado anterior, es decir, por sus vértices de mayor a menor z. Entonces, dadas dos caras a comparar:

- Será más alta la que tenga su primer vértice con mayor z
- Si éstos son iguales, la que tenga su segundo vértice con mayor z
- Si éstos también lo son, la que tenga su tercer vértice con mayor z
- Si suceden los tres casos anteriores, se elige como menor la cara que tenga menor superficie

En el caso de la figura, la ordenación devolvería (LTR2 LTR1 LTR0)

Colección Tareas de años pasados.

La representación más común de una matriz de ***mxn*** en LISP se logra mediante una lista de m sublistas, cada una de las cuales tiene n elementos. Según eso, el vector fila convencional se representa como **VF = ((VF_0 VF_1 . . VF_n))** -una lista de una sola sublista con n elementos- y el vector columna como **VC = ((VC_0) (VC_1) . . . (VC_m))** –una lista de m sublistas, cada una de ellas de un solo elemento.

1. Escribir una función **TRANSVF (VF / . . .)** que reciba un vector fila y devuelva su traspuesto. (0,5p)

2. Escribir una función **TRANSVC (VC / . . .)** que reciba un vector columna y devuelva su traspuesto. (0,5p)

3. Escribir una función VLISP **A*B (VF VC / …)** que devuelva el producto del vector fila VC por el vector columna VF. Se supone que los datos siempre llegan correctamente, es decir, no hace falta comprobar que la dimensión de los vectores es coherente. (1,5p)

4. Escribir una función VLISP **MA*MB (MA MB / . . .)** que devuelva la matriz resultado del producto de las matrices MA y MB. Se supone que los datos siempre llegan correctamente, es decir, no hace falta comprobar que la dimensión de los vectores es coherente. (4p)

Colección Tareas de años pasados.

4. Escribir una función VLISP **HIP1 (K N / . . .)** que devuelva una lista con los puntos de una hipérbola equilátera (ver ecuaciones y figuRA debajo) en el rango de ángulo Θ [44,0] (expresado en sexagesimanes). El argumento K es el parámetro de la hipérbola y el argumento N es el número de puntos que debe haber la lista solución, que será del tipo (P_0 P_1 . . . P_{N-1}) , correspondiendo los puntos extremos de la lista a los valores extremos del intervalo.

(1p)

Ecuación paramétrica de la hipérbola equilátera $X^2 - Y^2 = K^2$:

$X = K \bullet ch(\theta)$, $Y = K \bullet sh(\theta)$, donde $ch(\theta) = \dfrac{e^{\theta} + e^{-\theta}}{2}$, $sh(\theta) = \dfrac{e^{\theta} - e^{-\theta}}{2}$

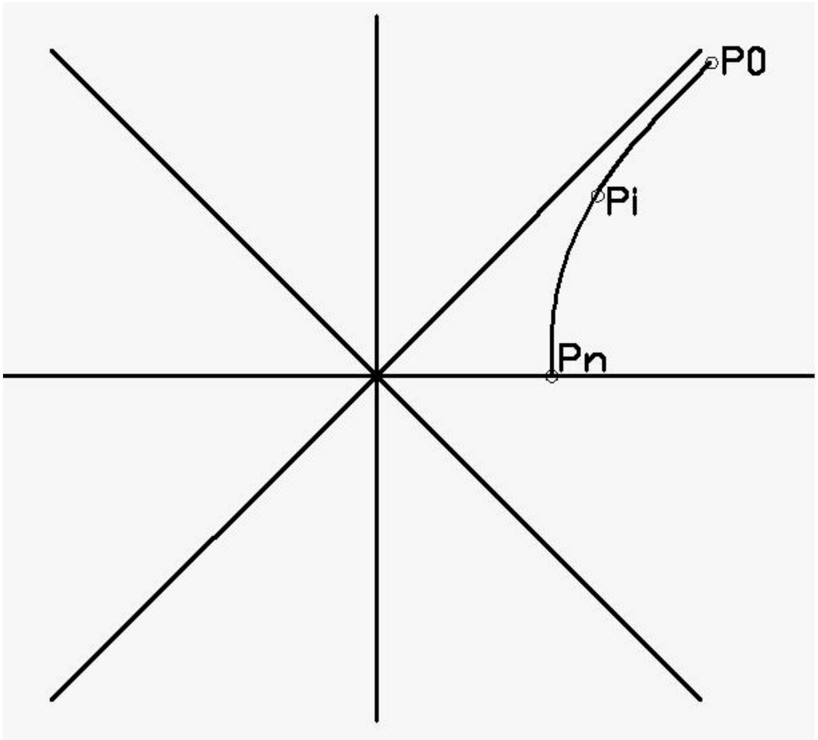

Figura 1. Medido desde el origen de coordenadas, el ángulo que determina el punto P0 es de 44º y va reduciéndose paulatinamente hasta llegar a 0º en el punto Pn.

Colección Tareas de años pasados.

5. Escribir una función Lisp **MEDIOS (LP / . . .)** que reciba una lista de puntos **lP = (P_0 P_1 . . P_n)** y devuelva la lista ***LPM = (PM_0 PM_1 . . PM_{n-1})*** de los puntos medios de los segmentos P_iP_{i+1}. La lista lp representa una poligonal abierta. (0,5P)

6. Escribir un función Lisp **UNION (LP LPM / . . .)** que devuelva la lista resultado **LUNION = (P_0 *PM_0* P_1 *PM_1. . PM_{n-1}* P_n)** . (1P)

7. Escribir una función Lisp **SIERRA (N P D / . . .)** que devuelva la lista de puntos **LQ = (Q_0 Q_1 . . Q_{2^n})**. La ley geométrica de formación de estos puntos **Q_i** se ofrece en la figura 1. (1,5P)

8. Sean dos listas, **LPP = (P_0 P_1 . . P_n)** y **LPQ = (Q_0 Q_1 . . Q_n)**, con el mismo número de puntos. Se desea escribir una función Lisp **COMBINA (LPP LPQ / …)** que devuelva la lista de puntos **LPR = (R_0 R_1 . . R_n)**, donde cada punto **R_i** tiene estas componentes: **(XP_i $YP_i+YQ_i-YP_0$ 0.0)** (0,5P)

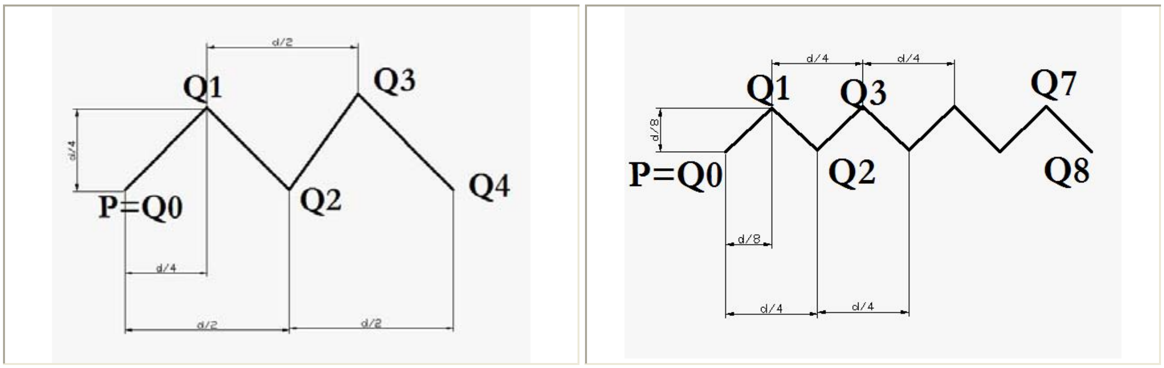

Figura 1. La poligonal que se obtiene con la función SIERRA tiene este aspecto. Se muestran los resultados para n=2 (izquierda) y n=3 (derecha). El valor de n nunca es menor que 2. En general, para n genérico, la lista **(Q_0 Q_1 . . $Q_{(2^n)}$)** tiene 2^n+1 elementos y las distancias acotadas entre elementos se deducen de la figura: en horizontal valen $d/2^{n-1}$, salvo la primera, que vale $d/2^n$; la vertical también vale $d/2^n$. Los puntos **Q_0 Q_2 . . . $Q_{(2^n)}$** están en una horizontal y los **Q_1 Q_3 . . . $Q_{(2^n-1)}$** también.