25 Es tracta de determinar els coeficients dels polinomis ortogonals de Chebyshev i els seus zeros.

Els polinomis de Chebyshev es poden generar de forma recurrent a partir de la següent fórmula:

$$\begin{aligned} T_0(x) &= 1, \\ T_1(x) &= x, \\ T_k(x) &= 2xT_{k-1}(x) - T_{k-2}(x), \forall k \geq 2. \end{aligned}$$

Les arrels o zeros d'aquests polinomis són nombres reals que estan a l'interval $[-1,1]$ i compleixen la propietat següent: les arrels de $T_{k-1}(x)$ separen les arrels de $T_k(x)$ $\forall k \geq 2$, és a dir, si $x_1^{k-1}, x_2^{k-1}, \ldots, x_{k-1}^{k-1}$ són les arrels de $T_{k-1}(x)$ i $x_1^k, x_2^k, \ldots, x_k^k$ són les arrels de $T_k(x)$, llavors tenim

$$-1 < x_1^k < x_1^{k-1} < x_2^k < x_2^{k-1} < \ldots < x_{k-1}^{k-1} < x_k^k < 1 \tag{1}$$

Els primers polinomis, els seus coeficients i les seves arrels són:

Polinomis	coeficients				arrels		
	a_0	a_1	a_2	a_3			
$T_0(x) = 1$	1						
$T_1(x) = x$	0	1			0		
$T_2(x) = 2x^2 - 1$	-1	0	2		$-1/\sqrt{2}$	$1/\sqrt{2}$	
$T_3(x) = 4x^3 - 3x$	0	-3	0	4	$-\sqrt{3}/2$	0	$\sqrt{3}/2$

Guardarem els coeficients dels polinomis en un vector de nom `coef` (c_i) i de dimensió $(n+2)(n+1)/2$ de la forma:

$$c_0 = 1,\ c_1 = 0,\ c_2 = 1,\ c_3 = -1,\ c_4 = 0,\ c_5 = 2,\ c_6 = 0,\ c_7 = -3,\ c_8 = 0,\ c_9 = 4, \ldots$$

De forma anàloga guardarem els zeros dels polinomis en un vector de nom `zeros` (z_i) de dimensió $(n+1)n/2$ de la forma:

$$z_0 = 0,\ z_1 = -1/\sqrt{2},\ z_2 = 1/\sqrt{2},\ z_3 = -\sqrt{3}/2,\ z_4 = 0,\ z_5 = \sqrt{3}/2, \ldots$$

a) La funció `main` llegirà el nom d'un fitxer d'entrada i d'un d'escriptura; del primer llegirà `n`, màxim grau dels polinomis que volem calcular, `itMax`, nombre màxim d'iteracions permès en el càlcul dels zeros, i `tol`, tolerància amb que volem calcular-los. Cridarà les funcions d'usuari necessàries i escriurà en el fitxer de sortida els coeficients dels polinomis i els seus zeros, seguint l'exemple que apareix a continuació. Guardeu aquesta funció en un fitxer de nom **pol_Ort_Main.c**.

b) Programeu les següents funcions d'usuari, guardant-les en un fitxer de nom **funcions_Polin.c**:

- `double horner(int n, double *p, double z)` que avalua un polinomi p de grau n en un punt z usant l'algorisme de Horner.

- `double *der_pol(int n, double *p)` que deriva polinomis de grau n, i retorna un apuntador al polinomi derivada de p. Feu-la de manera que derivi correctament també els polinomis de grau 0.

c) Programeu les següents funcions d'usuari, guardant-les en un fitxer de nom **pol_Ort_Fun.c**.

- `void recurrencia(int k, double *ultim, double *penultim, double *t_k)`
  Donats `k`, grau del polinomi a calcular, `ultim` i `penultim`, punters als coeficients de $T_{k-1}(x)$ i $T_{k-2}(x)$, respectivament, calcula els coeficients de $T_k(x)$, que guardarem a partir de `t_k`. Els coeficients dels polinomis es guardaran en l'ordre creixent dels monomis.
- `int newPol(int k, double *c, double x0, int itMax, double tol, double *zero)`
  Donats `k`, grau del polinomi, `c`, que apunta als coeficients de $T_k(x)$, i `x0`, valor aproximat d'una arrel de $T_k(x)$, la calcularem amb un error menor que `tol` usant el mètode de Newton, fent un màxim de `itMax` iteracions. Els valors del polinomi i de la seva derivada en un punt es calcularan usant l'algorisme de Horner. La funció retorna 1 si el mètode ha convergit; altrament retorna -1.
- `int calZeros(int k, double *c, int itMax, double tol, double *arr_ant, double *zeros)`
  Donats `k`, grau del polinomi, `c`, que apunta als coeficients de $T_k(x)$, `itMax`, nombre màxim d'iteracions que realitzarem, `tol`, tolerància amb que calcularem les arrels i `arr_ant`, vector que conté, ordenades de petites a grans, les arrels del polinomi $T_{k-1}(x)$, calcularà les arrels de $T_k(x)$ i les guardarà en el vector `zeros`. Per obtenir les arrels cridarem diverses vegades la funció `newPol` amb punt inicial, `x0`, el punt mig dels intervals descrits en (1). Retorna el valor -1, si hi ha hagut algun problema; altrament retorna 1.

Per exemple, per a $n = 3$, la sortida en el fitxer serà

```
Grau del polinomi = 0
T0[ 0] =   1.000000000000e+00

==================
Grau del polinomi = 1
T1[ 0] =   0.000000000000e+00
T1[ 1] =   1.000000000000e+00

ZT1[ 1] =  0.000000000000e+00
==================
Grau del polinomi =  2
T2[ 0] =  -1.000000000000e+00
T2[ 1] =   0.000000000000e+00
T2[ 2] =   2.000000000000e+00

ZT2[ 1] = -7.071067811865e-01
ZT2[ 2] =  7.071067811865e-01
==================
Grau del polinomi = 3
T3[ 0] =  -0.000000000000e+00
T3[ 1] =  -3.000000000000e+00
T3[ 2] =   0.000000000000e+00
T3[ 3] =   4.000000000000e+00

ZT3[ 1] = -8.660254037844e-01
ZT3[ 2] =  0.000000000000e+00
ZT3[ 3] =  8.660254037844e-01
==================
```