

```

/*
PROGRAMA PARA DERIVAR UN POLINOMIO

1. El usuario introduce los datos del polinomio, esto se hace en el metodo leerCoeficientes()
Polinomio que sigue esta forma:
P_n(x) = a_0 + a_1*x + a_2*x^2 + a_3*x^3 + ... + a_n*x^n
Donde n es el grado del polinomio.

2. Luego el usuario introduce el punto z.
Se realiza el cambio de base a la base (x-z). Para ello se divide el polinomio por (x-z).
Esto se realiza en el metodo divisionSintetica().

3. Se calcula las derivadas
      (i
      d P_n(x)
c (i) = -----
      (i
      d x          | x = z
*/

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

#define MAX 100

int main (int argc, char *argv[]) {
    /*prototype*/
    void leerCoeficientes(int, double *);
    void divisionSintetica(int, double, double *, double *);
    void calcularDerivadas(int, double *, double *);
    void escribirResultados(int, double, double *, double *, double *);
    /*variables*/
    int n; //polinomial's grade

    do{
        printf("\n Introduce el grado del polinomio (max=100) -> n = ");
        scanf("%d", &n);
    } while (n <= 0 || n >= 100);
    double a[n];
    double b[n];
    double c[n];

    double z;
    printf("\n Introduce el punto de calculo -> z = ");
    scanf("%lf", &z);

    /*call methods*/
    leerCoeficientes(n,a);
    divisionSintetica(n,z,a,b);
    calcularDerivadas(n,b,c);
    escribirResultados(n,z,a,b,c);

    /*wait for user keyboard*/
    fflush(stdin); getchar();

    return 0;
} //end of method main

/*metodo para leer de la pantalla los coeficientes introducidos por el usuario*/
void leerCoeficientes(int n, double v[]){

    printf("\n Introduzca los coeficientes del polinomio: ");
    int i;
    for(i=0; i <= n; i++){
        printf("\n\t a(%d) = ",i);
        scanf("%lf",&v[i]);
    }
    return ;
}

```

```

} //end of method leerCoeficientes

/*metodo para dividir el polinomio por (x-z)*/
void divisionSintetica(int n, double z, double a[], double b[]){
    int i;
    for(i = 0; i <= n; i++){
        b[i] = a[i];

        int k;
        for(k = n; k >= 1; k--){
            for(i = n-1; i >= n-k; i--){
                b[i] += b[i+1]*z;
            }
        }

        return ;
    } //end of method divisionSintetica

    /*metodo para dividir el polinomio por (x-z)*/
    void calcularDerivadas(int n, double a[], double b[]){
        int i;
        for(i = 0; i <= n; i++){
            b[i] = a[i];

            double fact = 1.;
            for(i = 2; i <= n; i++){
                fact *= (double)i;
                b[i] *= fact;
            }

            return ;
        } //end of method calcularDerivadas

        /*metodo para dividir el polinomio por (x-z)*/
        void escribirResultados(int n, double z, double a[], double b[], double c[]){
            printf("\n RESULTADOS");

            printf("\n Grado del polinomio -> n = %5d", n);
            printf("\n Punto de calculo -> z = %20.9f", z);

            printf("\n Coeficiente en la base {1,x,...,x^n}:");
            int i;
            for(i = 0; i <= n; i++){
                printf("\n\t a[%d] = %20.9f", i, a[i]);
            }

            printf("\n Coeficiente en la base {1,(x-z),...,(x-z)^n}:");
            for(i = 0; i <= n; i++){
                printf("\n\t a[%d] = %20.9f", i, b[i]);
            }

            printf("\n Funcion y derivadas en el punto x = z:");
            for(i = 0; i <= n; i++){
                printf("\n\t a[%d] = %20.9f", i, c[i]);
            }

            return ;
        } //end of method escribirResultados
    }
}

```