

```

#include <stdio.h>

int main (void)
{
    int ndv[3], k, i, s2, resto, dv1, dv2;
    double ncpf[9];
    float s1, dvfinal, dvfinaltemp[3];
    printf ("\nDigite o CPF(+Dig Verificador) na forma abcdefghi:");
    scanf ("%e",&ncpf);
    printf ("Digite o Digito verificador (xy):");
    scanf ("%i",&ndv);
    printf ("Digito ver digitado: %e-%i",ncpf,ndv);
    //Obtendo o primeiro digito verificador
    k=10; //Constante
    i=0; //identificador
    s1=0; //Limpendo buffer da var soma
    /* Para obter j multiplicamos a, b, c, d, e, f, g, h e i pelas
    * constantes correspondentes:
    * a x 10 . b x 9 . c x 8 . d x 7 . e x 6 . f x 5 . g x 4 . h x 3 . i x 2
    */
    for (; k>=2; k--, i++) {
        s1 = s1 + ncpf[i]*k;
        printf ("\n\ni=%i,\tk=%i\ns1=%f\n",i,k,s1);
    }
    /* O resultado da soma, S1 = 10a + 9b + 8c + 7d + 6e + 5f + 4g + 3h + 2i,
    * é dividido por 11 e o RESTO dessa divisão: RESTO.
    * Se for 0 ou 1
    * O dígito j é [0] (zero)
    * Se for 2, 3, 4, 5, 6, 7, 8, 9 ou 10
    * O dígito j é [11 - RESTO]
    */
    resto = s1%11;
    if (resto==0 || resto==1) dv1 = 0;
    if (resto>=2 && resto<=10) dv1 = 11-resto;
    //Obtendo o segundo digito verificador
    k=11;
    i=0;
    s2=0;
    /* Para obter k multiplicamos a, b, c, d, e, f, g, h, i e j
    * pelas constantes correspondentes:
    * a x 11 . b x 10 . c x 9 . d x 8 . e x 7 . f x 6 . g x 5 . h x 4 . i x 3 . j x 2
    */
    for (; k>=3; k--, i++){
        s2 = s2 + ncpf[i]*k;
    }
    s2 = s2 + 2*dv1; /*Obtendo o restante da somatoria
    * obs: não entrou no laço pq dv1 está em outro vetor
    */
    /* O resultado da soma,
    * S2 = 11a + 10b + 9c + 8d + 7e + 6f + 5g + 4h + 3i + 2j,
    * é dividido por 11 e o RESTO dessa divisão:
    * Se for 0 ou 1, O dígito k é [0] (zero)
    * Se for 2, 3, 4, 5, 6, 7, 8, 9 ou 10
    * O dígito k é [11 - RESTO]
    */
    resto=0; //Limpendo valores da var temporaria resto
    resto = s2%11;
    if (resto==0 || resto==1) dv2=0;
    if (resto>=2 && resto<=10) dv2=11-resto;
    dvfinal = dv1*10.0 + dv2;
    if (ndv==dvfinal) printf ("\nO CPF %e-%.0f eh valido",ncpf, dvfinal);
    else printf ("\nO CPF %f-%.0i eh invalido",ncpf, ndv);
}

```