

```
#INCLUDE <STDIO.H> // BIBLIOTECA DE ENTRADA E SAÍDA DE DADOS.  
#INCLUDE <STRING.H> // BIBLIOTECA DE MANIPULAÇÃO DE STRING
```

```
INT MAIN() { // FUNÇÃO QUE IRÁ RODAR CÓDIGO.
```

```
    CHAR RGM[9]; // ARRAY DE 8 DÍGITOS
```

```
    INT I, J, TEMP; // VARIÁVEIS NO FORMATO INTEIRO
```

```
    PRINTF("DIGITE SEU RGM (8 DÍGITOS): "); // MOSTRA PRO USUÁRIO
```

```
    SCANF("%8s", RGM); // RECEBE O QUE O USUÁRIO DE DIGITA E ARMA-
```

```
    // ZENA NA VARIÁVEL RGM
```

```
    IF (STRLEN != 8) { // CONDIÇÃO PARA VER SE NÃO TEM 8 DÍGITOS.
```

```
        PRINTF("ERRO: RGM DEVE TER 8 DÍGITOS."); // MOSTRA PRO USUÁRIO.
```

```
        RETURN 1; // RETORNA UM CÓDIGO DE ERRO;
```

```
    }
```

```
    FOR (I=0; I<7; I++) {
```

```
        FOR (J=0; J<7-I; J++) {
```

```
            IF (RGM[J] > RGM RGM[J+1]) {
```

```
                TEMP = RGM[J];
```

```
                RGM[J] = RGM[J+1];
```

```
                RGM[J+1] = TEMP;
```

```
            }
```

```
        }
```

```
    }
```

```
    // I É O NÚMERO, O CÓDIGO USA UM '-' E O + PARA MUDAR AS POSIÇÕES
```

```
    // DOS NÚMEROS, ASSIM DEIXANDO ELE EM ORDEM CRESCENTE, O TEMP É UMA VARIÁVEL
```

```
    // PARA AJUDAR NESSA ORGANIZAÇÃO, O TEMP SE IGUALA COM O RGM PARA DEIXAR
```

```
    // CERTINHO.
```



PRINTF ("RGM ORDENADO EM ORDEM CRESCENTE: %S \n", RGM); // MOSTRA

// PARA O USUÁRIO O RGM EM ORDEM CRESCENTE, E QUEBRA UMA LINHA;

~~RETORNANDO O RGM PARA O USUÁRIO E MANDANDO O RGM PARA O PROGRAMA QUE ACABOU.~~

RETURN 0; // MANDA PRO PROGRAMA QUE ACABOU. }

DIGITE SEU RGM (8 DÍGITOS): 45620792

RGM ORDENADO EM ORDEM CRESCENTE: 02245679