```
40  static void create_dct_matrix(FLOAT filter[16][32])
41  {
42      register int i, k;
43      register int aux = 0;
44
45  #ifdef FLOAT_DOUBLE
46      for (i = 0; i < 16; i++)
47          for (k = 0; k < 32; k++) {
48              //if ((filter[i][k] = 1e9 * cos((FLOAT) ((2 * i + 1) * k * PI64))) >= 0)
49              if ((filter[i][k] = tabcos_create_dct_matrix[aux++]) >= 0)
50                  modf(filter[i][k] + 0.5, &filter[i][k]);
51              else
52                  modf(filter[i][k] - 0.5, &filter[i][k]);
53              filter[i][k] *= 1e-9;
54          }
55  #else
56      for (i = 0; i < 16; i++)
57          for (k = 0; k < 32; k++) {
58              //if ((filter[i][k] = 1e9 * cos((FLOAT) ((2 * i + 1) * k * PI64))) >= 0)
59              if ((filter[i][k] = tabcos_create_dct_matrix[aux++]) >= 0)
60                  modff(filter[i][k] + 0.5, &filter[i][k]);
61              else
62                  modff(filter[i][k] - 0.5, &filter[i][k]);
63              filter[i][k] *= 1e-9;
64          }
65  #endif
66  }
```