

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  #define N 5 /* 要素数の指定 */
5
6  double *dvector(int i, int j); /* ベクトル領域の確保 */
7  void free_dvector(double *a, int i); /* 領域の解放 */
8
9  int main(void)
10 {
11     double *a, *b;
12     int i;
13
14     printf("a の領域確保\n"); /* 配列 a の添字は 1~N */
15     a = dvector(1,N);
16     for(i = 1; i <= N; i++){
17         a[i] = (double)i / 20.0;
18         printf("a[%d] = %f \n", i, a[i]);
19     }
20     free_dvector(a,1);
21     printf("aを解放しました\n");
22
23     printf("bの領域確保\n");
24     b = dvector(2,N+1); /* 配列 b の添字は 2~N+1 */
25     for(i = 2; i <= N+1; i++){
26         b[i] = (double)i / 20.0;
27         printf("b[%d] = %f \n", i, b[i]);
28     }
29     free_dvector(b,2);
30     printf("b を解放しました\n");
31
32     return 0;
33 }
34
35 double *dvector(int i, int j) /* a[i]~a[j] の領域を確保 */
36 {
37     double *a;
38
39     if((a = malloc( ((j - i + 1) * sizeof(double))) ) == NULL)
40     {
41         printf("メモリが確保できません (from dvector) \n");
42         exit(1);
43     }
44
45     return (a - i);
46 }
47
48 void free_dvector(double *a, int i)
49 {
50     free( (void *) (a + i) ); /* (void *) 型へのキャストが必要 */
51 }

```