```
 1: #include <stdio.h>
 2: #include <stdlib.h>
 3:
 4: int dead_compute01( int *a, int size){
 5:
 6:    int j=0 ;
 7:    long count=0;
 8:
 9:    for (j=0; j<size*3; j++){
10:
11:     int i, tmp=0;
12:
13:     for (i = 0; i<size; i++){
14:         a[i] = i*i-tmp;
15:             tmp += i;
16:     }
17:
18:     printf("%s: %d\t", __FUNCTION__, a[(i+j)/4]);
19:     if(j%10 == 0) printf("\n");
20:
21:     for (i =0 ; i< size; i++){
22:         a[i] = i*i/2;
23:                 tmp = tmp + 2*i -200;
24:     }
25:
26:     count += tmp * 2 - tmp / 3;
27:
28:     }
29:     return 0;
30: }
31:
32: int main(){
33:     int size = 2000;
34:     int *a = malloc (size * sizeof(int));
35:     dead_compute01( a, size);
36:     int i;
37:     for (i = size-1; i>=0; i--){
38:         printf("a[%d]=%d\t", i, a[i]);
39:         if(i%10==0) printf("\n");
40:     }
41:     free(a);
42:     return 0;
43: }
```