```
#include <stdio.h>
#include <stdlib.h>

#define TAM 32

int calcula(unsigned n)
{     int c;
      unsigned vector[TAM], result=0;
      for (c = 0; c < TAM; c++) {
      vector[c]= rand()%n + 1;
      // printf("%d :", vector[c]);
      }
     for (c = 0; c < TAM; c++)
       result+=vector[c];
     return result/TAM;
}


int main() {
  unsigned n;

  printf("Introduza gama de valores [1,N]\n");
  scanf("%u", &n);

  printf("\n Media: %d \n", calcula (n));
  return 0;
}
```

```
--executável com -O0 após desmontagem em gdb--

...

0x080483f4 <calcula>:
0x080483f4:  push   %ebp
0x080483f5:  mov    %esp,%ebp
0x080483f7:  push   %ebx
0x080483f8:  sub    $0xa4,%esp
0x080483fe:  movl   $0x0,-0x9c(%ebp)
0x08048408:  movl   $0x0,-0xc(%ebp)
0x0804840f:  cmpl   $0x1f,-0xc(%ebp)
0x08048413:  jle    0x8048417
0x08048415:  jmp    0x8048438
0x08048417:  mov    -0xc(%ebp),%ebx
0x0804841a:  call   0x8048328 <rand@plt>
0x0804841f:  mov    $0x0,%edx
0x08048424:  divl   0x8(%ebp)
0x08048427:  lea    0x1(%edx),%eax
0x0804842a:  mov    %eax,-0x98(%ebp,%ebx,4)
0x08048431:  lea    -0xc(%ebp),%eax
0x08048434:  incl   (%eax)
0x08048436:  jmp    0x804840f
0x08048438:  movl   $0x0,-0xc(%ebp)
0x0804843f:  cmpl   $0x1f,-0xc(%ebp)
0x08048443:  jle    0x8048447
0x08048445:  jmp    0x8048460
0x08048447:  mov    -0xc(%ebp),%eax
0x0804844a:  mov    -0x98(%ebp,%eax,4),%edx
0x08048451:  lea    -0x9c(%ebp),%eax
0x08048457:  add    %edx,(%eax)
0x08048459:  lea    -0xc(%ebp),%eax
0x0804845c:  incl   (%eax)
0x0804845e:  jmp    0x804843f
0x08048460:  mov    -0x9c(%ebp),%eax
0x08048466:  shr    $0x5,%eax
0x08048469:  add    $0xa4,%esp
0x0804846f:  pop    %ebx
0x08048470:  leave
0x08048471:  ret
```

```
--executável com –O2 após desmontagem em gdb--
...
08048424 <calcula>:
 8048424: 55                  push %ebp
 8048425: 89 e5               mov  %esp,%ebp
 8048427: 57                  push %edi
 8048428: 56                  push %esi
 8048429: 53                  push %ebx
 804842a: 81 ec 8c 00 00 00   sub  $0x8c,%esp
 8048430: 8b 7d 08            mov  0x8(%ebp),%edi
 8048433: 31 f6               xor  %esi,%esi
 8048435: 31 db               xor  %ebx,%ebx
 8048437: 90                  nop
 8048438: e8 1b ff ff ff      call 8048358 <rand@plt>
 804843d: 31 d2               xor  %edx,%edx
 804843f: f7 f7               div  %edi
 8048441: 42                  inc  %edx
 8048442: 89 94 9d 68 ff ff ff mov %edx,-0x98(ebp,ebx,4)
 8048449: 43                  inc  %ebx
 804844a: 83 fb 1f            cmp  $0x1f,%ebx
 804844d: 7e e9               jle  8048438
 804844f: 31 db               xor  %ebx,%ebx
 8048451: 8d 76 00            lea  0x0(%esi),%esi
 8048454: 03 b4 9d 68 ff ff ff add -0x98(ebp,ebx,4),%esi
 804845b: 43                  inc  %ebx
 804845c: 83 fb 1f            cmp  $0x1f,%ebx
 804845f: 7e ??               jle  8048454
 8048461: 81 c4 8c 00 00 00   add  $0x8c,%esp
 8048467: c1 ee 05            shr  $0x5,%esi
 804846a: 5b                  pop  %ebx
 804846b: 89 f0               mov  %esi,%eax
 804846d: 5e                  pop  %esi
 804846e: 5f                  pop  %edi
 804846f: c9                  leave
 8048470: c3                  ret
```

```
---- execução do programa com breakpoints ----
---- (executável com -O2) --------------------
...
(gdb) run
Starting program: ...
Introduza a gama de valores [1,N]
20
---------- paragem num breakpoint ----------
(gdb) info reg
eax            0x55ecd2e        90098990
ecx            0x7d4308         8209160
edx            0x3              3
ebx            0x0              0
esp            0xbfffe9a0       0xbfffe9a0
ebp            0xbfffea38       0xbfffea38
esi            0x0              0
edi            0x14             20
eip            0x8048451        0x8048451
eflags         0x246            [ PF ZF IF ]
...

(gdb) x/36xw $ebp-0xa0
0xbfffe998:  0xbfffea38    0x0804843d
             0x00000004    0x00000007
0xbfffe9a8:  0x00000012    0x00000010
             0x0000000e    0x00000010
0xbfffe9b8:  0x00000007    0x0000000d
             0x0000000a    0x00000002
0xbfffe9c8:  0x00000003    0x00000008
             0x0000000b    0x00000014
0xbfffe9d8:  0x00000004    0x00000007
             0x00000001    0x00000007
0xbfffe9e8:  0x0000000d    0x00000011
             0x0000000c    0x00000009
0xbfffe9f8:  0x00000008    0x0000000a
             0x00000003    0x0000000b
0xbfffea08:  0x00000003    0x00000004
             0x00000008    0x00000010
0xbfffea18:  0x0000000a    0x00000003
             0x007d4420    0x080485e0

(gdb) x/6xw $ebp-12
0xbfffea2c:  0x007d3ff4    0x00573ca0
             0x00000000    0xbfffea58
0xbfffea3c:  0x080484a3    0x00000014
```