

```

1  heap layout for x86
2
3      <-- call_helper() function stack base, %ebp, points to
4  local variables | //..... |
5      <-->
6      | thread |
7      <-- 36(%ebp) (9 * wordSize)
8      | parameterSize |
9      <-- 32(%ebp) (8 * wordSize)
10     | parameters |
11     <-- 28(%ebp) (7 * wordSize)
12 Input param-- | entry_point |
13     <-- 24(%ebp) (6 * wordSize)
14     | method |
15     <-- 20(%ebp) (5 * wordSize)
16     | result_type |
17     <-- 16(%ebp) (4 * wordSize)
18     | resultAddress |
19     <-- 12(%ebp) (3 * wordSize)
20     | link |
21     <-- 8(%ebp)
22 4 bytes | eip |
23 4 bytes | ebp |
24 4 bytes |
25     <-- CallStub() function stack base, %ebp, points to
26     | edi |
27 16 bytes-- | esi |
28     <-->
29     | ebx |
30     <-->
31     | mxcsr |
32     <-->
33     <-->
34     | arg1 | <--> | arg1 |
35     <-- 8(%esp)
36     | arg2 | <--> | arg2 |
37     <-- 4(%esp)
38     | arg3 | <--> | arg3 |
39     <-- (%esp) -- CallStub()
40     | eip |
41     <-- The address of next assembly code in CallStub (after call *%eax)
42

```

Java function input memory layout