Halting problem undecidability and infinitely nested simulation

When halting is defined as any computation that halts without ever having its simulation aborted then it can be understood that partial halt decider H correctly decides not halting on the simplified version of the Linz Ĥ. When this simplified concrete example is fully understood then the exact same reasoning can be applied to the actual Linz H correctly deciding not halting on its input.

The x86utm operating system was created so that the halting problem could be examined concretely in the high level language of C. x86utm UTM tape elements are 32-bit unsigned integers. H examines the behavior of the x86 emulation of its input. As soon as a non-halting behavior pattern is matched H aborts the simulation of its input and decides not halting.

```c
int Simulate(u32 P, u32 I) {
  ((void(*)(u32))P)(I);
  return 1;
}


// Simplified Linz Ĥ (Linz:1990:319)
void H_Hat(u32 P) {
  // Linz H as a simulating partial halt decider
  u32 Input_Halts = H(P, P);
  if (Input_Halts)
    HERE: goto HERE;
}


void H_Hat2(u32 P) {
  u32 Input_Halts = Simulate(P, P);
  if (Input_Halts)
    HERE: goto HERE;
}


int main() {
  H_Hat2((u32)H_Hat2);
  H_Hat((u32)H_Hat);
}
```

Anyone that knows C programming very well will know that line 1 of main() won't halt and line 2 of main() will only halt if simulating partial halt decider H() stops simulating H_Hat(). A simulating halt decider that never stops simulating its input is simply a simulator on this input.

When we know that the UTM simulation of TM Description P on input I would never halt we know that the execution of TM P(I) would never halt.

On this basis we know that any computation that must have its simulation aborted to prevent its otherwise infinite execution is correctly rejected as non-halting.

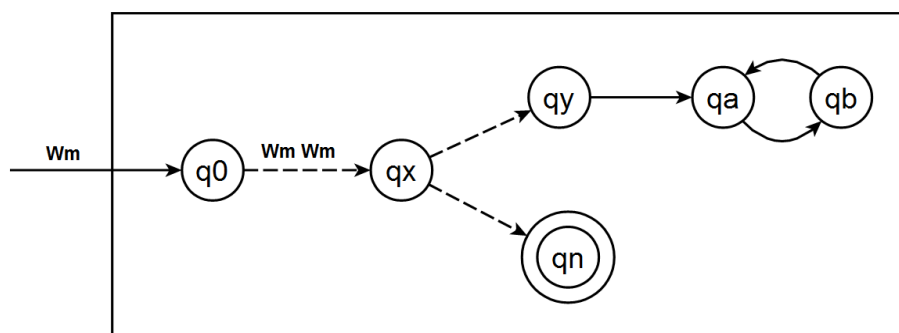# Peter Linz Ĥ applied to the Turing machine description of itself: [Ĥ]

Figure 12.3 Turing Machine Ĥ

Ĥ.q0 wM ⊢* Ĥ.qx wM wM ⊢* Ĥ.qy ∞
Ĥ.q0 wM ⊢* Ĥ.qx wM wM ⊢* Ĥ.qn

The above is adapted from (Linz:1990:319).
It shows that Turing machine Ĥ copies its input at (q0) and begins executing an embedded copy of the original halt decider with this input at (qx).

The (qy) state indicates that the halt decider has determined that its input would halt. The ((qn)) state indicates the input would not halt. The appended (qa) and (qb) states cause Ĥ to infinitely loop if the halt decider decides that its input would halt.

The above definition specifies this execution trace:
It can be understood from the above specification that when the embedded halt decider at Ĥ.qx bases its halting decision on simulating its input, and it has ([Ĥ],[Ĥ]) as its input that:
- Ĥ.q0 would copy its input and then Ĥ.qx would simulate its input with this copy then
- Ĥ.q0 would copy its input and then Ĥ.qx would simulate its input with this copy then
- Ĥ.q0 would copy its input and then Ĥ.qx would simulate its input with this copy...

unless and until the halt decider at Ĥ.qx stops simulating its input.

When halting is defined as any computation that halts without ever having its simulation aborted by a simulating halt decider then we can see that the halt decider at state Ĥ.qx stops simulating its input and correctly transitions to its final Ĥ.qn state deciding not halting on its input in the computation: Ĥ((Ĥ)).

When halting is defined as any computation that halts without ever having its simulation aborted then the fact that Ĥ((Ĥ)) stops running does not indicate that it is a halting computation. The simulating halt decider would also force an infinite loop to stop running.

Linz, Peter 1990. An Introduction to Formal Languages and Automata. Lexington/Toronto: D. C. Heath and Company.

Appendix   (not needed to understand the proof)

```c
int Factorial(int n) {
  Output("Factorial(n)",n);
  if (n > 1)
    return n * Factorial(n - 1);
  else
    return 1;
}


void Infinite_Loop() {
  HERE: goto HERE;
}


int Infinite_Recursion(u32 N) {
  u32 M = Infinite_Recursion(N);
  return M;
}


int Simulate(u32 P, u32 I) {
  ((int(*)(int))P)(I);
  return 1;
}


void H_Hat2(u32 P) {
  u32 Input_Halts = Simulate(P, P);
  if (Input_Halts)
    HERE: goto HERE;
}


void H_Hat(u32 P) {
  u32 Input_Halts = H(P, P);
  if (Input_Halts)
    HERE: goto HERE;
}


int main() {
  u32 Input_Would_Halt1 = H((u32)Factorial, 3);
  Output("Input_Would_Halt1 = ", Input_Would_Halt1);

  u32 Input_Would_Halt2 = H((u32)Infinite_Loop, (u32)Infinite_Loop);
  Output("Input_Would_Halt2 = ", Input_Would_Halt2);

  u32 Input_Would_Halt3 = H((u32)Infinite_Recursion, 3);
  Output("Input_Would_Halt3 = ", Input_Would_Halt3);

  u32 Input_Would_Halt4 = H((u32)H_Hat2, (u32)H_Hat2);
  Output("Input_Would_Halt4 = ", Input_Would_Halt4);

  u32 Input_Would_Halt5 = Simulate((u32)H_Hat, (u32)H_Hat);
  Output("Input_Would_Halt5 = ", Input_Would_Halt5);

  u32 Input_Would_Halt6 = H((u32)H_Hat, (u32)H_Hat);
  Output("Input_Would_Halt6 = ", Input_Would_Halt6);

  H_Hat((u32)H_Hat);
}
```

```
_Factorial()
[00000ac8](01)    55                push ebp
[00000ac9](02)    8bec              mov ebp,esp
[00000acb](03)    8b4508            mov eax,[ebp+08]
[00000ace](01)    50                push eax
[00000acf](05)    681b030000        push 0000031b
[00000ad4](05)    e8fff8ffff        call 000003d8
[00000ad9](03)    83c408            add esp,+08
[00000adc](04)    837d0801          cmp dword [ebp+08],+01
[00000ae0](02)    7e17              jng 00000af9
[00000ae2](03)    8b4d08            mov ecx,[ebp+08]
[00000ae5](03)    83e901            sub ecx,+01
[00000ae8](01)    51                push ecx
[00000ae9](05)    e8daffffff        call 00000ac8
[00000aee](03)    83c404            add esp,+04
[00000af1](04)    0faf4508          imul eax,[ebp+08]
[00000af5](02)    eb07              jmp 00000afe
[00000af7](02)    eb05              jmp 00000afe
[00000af9](05)    b801000000        mov eax,00000001
[00000afe](01)    5d                pop ebp
[00000aff](01)    c3                ret
Size in bytes:(0056) [00000aff]

_Infinite_Loop()
[00000b08](01)    55                push ebp
[00000b09](02)    8bec              mov ebp,esp
[00000b0b](02)    ebfe              jmp 00000b0b
[00000b0d](01)    5d                pop ebp
[00000b0e](01)    c3                ret
Size in bytes:(0007) [00000b0e]

_Infinite_Recursion()
[00000b18](01)    55                push ebp
[00000b19](02)    8bec              mov ebp,esp
[00000b1b](01)    51                push ecx
[00000b1c](03)    8b4508            mov eax,[ebp+08]
[00000b1f](01)    50                push eax
[00000b20](05)    e8f3ffffff        call 00000b18
[00000b25](03)    83c404            add esp,+04
[00000b28](03)    8945fc            mov [ebp-04],eax
[00000b2b](03)    8b45fc            mov eax,[ebp-04]
[00000b2e](02)    8be5              mov esp,ebp
[00000b30](01)    5d                pop ebp
[00000b31](01)    c3                ret
Size in bytes:(0026) [00000b31]

_Simulate()
[00000b38](01)    55                push ebp
[00000b39](02)    8bec              mov ebp,esp
[00000b3b](03)    8b450c            mov eax,[ebp+0c]
[00000b3e](01)    50                push eax
[00000b3f](03)    ff5508            call dword [ebp+08]
[00000b42](03)    83c404            add esp,+04
[00000b45](05)    b801000000        mov eax,00000001
[00000b4a](01)    5d                pop ebp
[00000b4b](01)    c3                ret
Size in bytes:(0020) [00000b4b]
```

```
_H_Hat2()
[00000b58](01)   55              push ebp
[00000b59](02)   8bec            mov ebp,esp
[00000b5b](01)   51              push ecx
[00000b5c](03)   8b4508          mov eax,[ebp+08]
[00000b5f](01)   50              push eax
[00000b60](03)   8b4d08          mov ecx,[ebp+08]
[00000b63](01)   51              push ecx
[00000b64](05)   e8cfffffff      call 00000b38
[00000b69](03)   83c408          add esp,+08
[00000b6c](03)   8945fc          mov [ebp-04],eax
[00000b6f](04)   837dfc00        cmp dword [ebp-04],+00
[00000b73](02)   7402            jz 00000b77
[00000b75](02)   ebfe            jmp 00000b75
[00000b77](02)   8be5            mov esp,ebp
[00000b79](01)   5d              pop ebp
[00000b7a](01)   c3              ret
Size in bytes:(0035) [00000b7a]

_H_Hat()
[00000b88](01)   55              push ebp
[00000b89](02)   8bec            mov ebp,esp
[00000b8b](01)   51              push ecx
[00000b8c](03)   8b4508          mov eax,[ebp+08]
[00000b8f](01)   50              push eax
[00000b90](03)   8b4d08          mov ecx,[ebp+08]
[00000b93](01)   51              push ecx
[00000b94](05)   e81ffeffff      call 000009b8
[00000b99](03)   83c408          add esp,+08
[00000b9c](03)   8945fc          mov [ebp-04],eax
[00000b9f](04)   837dfc00        cmp dword [ebp-04],+00
[00000ba3](02)   7402            jz 00000ba7
[00000ba5](02)   ebfe            jmp 00000ba5
[00000ba7](02)   8be5            mov esp,ebp
[00000ba9](01)   5d              pop ebp
[00000baa](01)   c3              ret
Size in bytes:(0035) [00000baa]

_main()
[00000bb8](01)   55              push ebp
[00000bb9](02)   8bec            mov ebp,esp
[00000bbb](03)   83ec18          sub esp,+18
[00000bbe](02)   6a03            push +03
[00000bc0](05)   68c80a0000      push 00000ac8
[00000bc5](05)   e8eefdffff      call 000009b8
[00000bca](03)   83c408          add esp,+08
[00000bcd](03)   8945fc          mov [ebp-04],eax
[00000bd0](03)   8b45fc          mov eax,[ebp-04]
[00000bd3](01)   50              push eax
[00000bd4](05)   682b030000      push 0000032b
[00000bd9](05)   e8faf7ffff      call 000003d8
[00000bde](03)   83c408          add esp,+08
[00000be1](05)   68080b0000      push 00000b08
[00000be6](05)   68080b0000      push 00000b08
[00000beb](05)   e8c8fdffff      call 000009b8
[00000bf0](03)   83c408          add esp,+08
[00000bf3](03)   8945f8          mov [ebp-08],eax
[00000bf6](03)   8b4df8          mov ecx,[ebp-08]
[00000bf9](01)   51              push ecx
[00000bfa](05)   6843030000      push 00000343
[00000bff](05)   e8d4f7ffff      call 000003d8
[00000c04](03)   83c408          add esp,+08
[00000c07](02)   6a03            push +03
[00000c09](05)   68180b0000      push 00000b18
[00000c0e](05)   e8a5fdffff      call 000009b8
[00000c13](03)   83c408          add esp,+08
```

2021-06-05   09:55 AM

```
[00000c16](03)      8945f4            mov [ebp-0c],eax
[00000c19](03)      8b55f4            mov edx,[ebp-0c]
[00000c1c](01)      52                push edx
[00000c1d](05)      685b030000        push 0000035b
[00000c22](05)      e8b1f7ffff        call 000003d8
[00000c27](03)      83c408            add esp,+08
[00000c2a](05)      68580b0000        push 00000b58
[00000c2f](05)      68580b0000        push 00000b58
[00000c34](05)      e87ffdffff        call 000009b8
[00000c39](03)      83c408            add esp,+08
[00000c3c](03)      8945f0            mov [ebp-10],eax
[00000c3f](03)      8b45f0            mov eax,[ebp-10]
[00000c42](01)      50                push eax
[00000c43](05)      6873030000        push 00000373
[00000c48](05)      e88bf7ffff        call 000003d8
[00000c4d](03)      83c408            add esp,+08
[00000c50](05)      68880b0000        push 00000b88
[00000c55](05)      68880b0000        push 00000b88
[00000c5a](05)      e8d9feffff        call 00000b38
[00000c5f](03)      83c408            add esp,+08
[00000c62](03)      8945ec            mov [ebp-14],eax
[00000c65](03)      8b4dec            mov ecx,[ebp-14]
[00000c68](01)      51                push ecx
[00000c69](05)      688b030000        push 0000038b
[00000c6e](05)      e865f7ffff        call 000003d8
[00000c73](03)      83c408            add esp,+08
[00000c76](05)      68880b0000        push 00000b88
[00000c7b](05)      68880b0000        push 00000b88
[00000c80](05)      e833fdffff        call 000009b8
[00000c85](03)      83c408            add esp,+08
[00000c88](03)      8945e8            mov [ebp-18],eax
[00000c8b](03)      8b55e8            mov edx,[ebp-18]
[00000c8e](01)      52                push edx
[00000c8f](05)      68a3030000        push 000003a3
[00000c94](05)      e83ff7ffff        call 000003d8
[00000c99](03)      83c408            add esp,+08
[00000c9c](05)      68880b0000        push 00000b88
[00000ca1](05)      e82ffeffff        call 00000b88
[00000ca6](03)      83c404            add esp,+04
[00000ca9](02)      33c0              xor eax,eax
[00000cab](02)      8be5              mov esp,ebp
[00000cad](01)      5d                pop ebp
[00000cae](01)      c3                ret
Size in bytes:(0247) [00000cae]


================================
...[00000bb8][001017c6][00000000](01)   55          push ebp
...[00000bb9][001017c6][00000000](02)   8bec        mov ebp,esp
...[00000bbb][001017ae][90909090](03)   83ec18      sub esp,+18
...[00000bbe][001017aa][00000003](02)   6a03        push +03
...[00000bc0][001017a6][00000ac8](05)   68c80a0000  push 00000ac8
...[00000bc5][001017a2][00000bca](05)   e8eefdffff  call 000009b8
Begin Local Halt Decider Simulation at Machine Address:ac8
...[00000ac8][00211866][0021186a](01)   55          push ebp
...[00000ac9][00211866][0021186a](02)   8bec        mov ebp,esp
...[00000acb][00211866][0021186a](03)   8b4508      mov eax,[ebp+08]
...[00000ace][00211862][00000003](01)   50          push eax
...[00000acf][0021185e][0000031b](05)   681b030000  push 0000031b
---[00000ad4][0021185e][0000031b](05)   e8fff8ffff  call 000003d8
Factorial(n)3
...[00000ad9][00211866][0021186a](03)   83c408      add esp,+08
...[00000adc][00211866][0021186a](04)   837d0801    cmp dword [ebp+08],+01
...[00000ae0][00211866][0021186a](02)   7e17        jng 00000af9
...[00000ae2][00211866][0021186a](03)   8b4d08      mov ecx,[ebp+08]
...[00000ae5][00211866][0021186a](03)   83e901      sub ecx,+01
...[00000ae8][00211862][00000002](01)   51          push ecx
```

2021-06-05   09:55 AM

```
...[00000ae9][0021185e][00000aee](05)    e8daffffff        call 00000ac8
...[00000ac8][0021185a][00211866](01)    55                push ebp
...[00000ac9][0021185a][00211866](02)    8bec              mov ebp,esp
...[00000acb][0021185a][00211866](03)    8b4508            mov eax,[ebp+08]
...[00000ace][00211856][00000002](01)    50                push eax
...[00000acf][00211852][0000031b](05)    681b030000        push 0000031b
---[00000ad4][00211852][0000031b](05)    e8fff8ffff        call 000003d8
Factorial(n)2
...[00000ad9][0021185a][00211866](03)    83c408            add esp,+08
...[00000adc][0021185a][00211866](04)    837d0801          cmp dword [ebp+08],+01
...[00000ae0][0021185a][00211866](02)    7e17              jng 00000af9
...[00000ae2][0021185a][00211866](03)    8b4d08            mov ecx,[ebp+08]
...[00000ae5][0021185a][00211866](03)    83e901            sub ecx,+01
...[00000ae8][00211856][00000001](01)    51                push ecx
...[00000ae9][00211852][00000aee](05)    e8daffffff        call 00000ac8
...[00000ac8][0021184e][0021185a](01)    55                push ebp
...[00000ac9][0021184e][0021185a](02)    8bec              mov ebp,esp
...[00000acb][0021184e][0021185a](03)    8b4508            mov eax,[ebp+08]
...[00000ace][0021184a][00000001](01)    50                push eax
...[00000acf][00211846][0000031b](05)    681b030000        push 0000031b
---[00000ad4][00211846][0000031b](05)    e8fff8ffff        call 000003d8
Factorial(n)1
...[00000ad9][0021184e][0021185a](03)    83c408            add esp,+08
...[00000adc][0021184e][0021185a](04)    837d0801          cmp dword [ebp+08],+01
...[00000ae0][0021184e][0021185a](02)    7e17              jng 00000af9
...[00000af9][0021184e][0021185a](05)    b801000000        mov eax,00000001
...[00000afe][00211852][00000aee](01)    5d                pop ebp
...[00000aff][00211856][00000001](01)    c3                ret
...[00000bca][001017ae][90909090](03)    83c408            add esp,+08
...[00000bcd][001017ae][90909090](03)    8945fc            mov [ebp-04],eax
...[00000bd0][001017ae][90909090](03)    8b45fc            mov eax,[ebp-04]
...[00000bd3][001017aa][00000001](01)    50                push eax
...[00000bd4][001017a6][0000032b](05)    682b030000        push 0000032b
---[00000bd9][001017a6][0000032b](05)    e8faf7ffff        call 000003d8
Input_Would_Halt1 = 1
...[00000bde][001017ae][90909090](03)    83c408            add esp,+08
...[00000be1][001017aa][00000b08](05)    68080b0000        push 00000b08
...[00000be6][001017a6][00000b08](05)    68080b0000        push 00000b08
...[00000beb][001017a2][00000bf0](05)    e8c8fdffff        call 000009b8
Begin Local Halt Decider Simulation at Machine Address:b08
...[00000b08][0025c28e][0025c292](01)    55                push ebp
...[00000b09][0025c28e][0025c292](02)    8bec              mov ebp,esp
...[00000b0b][0025c28e][0025c292](02)    ebfe              jmp 00000b0b
...[00000b0b][0025c28e][0025c292](02)    ebfe              jmp 00000b0b
Local Halt Decider: Infinite Loop Detected Simulation Stopped
...[00000bf0][001017ae][90909090](03)    83c408            add esp,+08
...[00000bf3][001017ae][90909090](03)    8945f8            mov [ebp-08],eax
...[00000bf6][001017ae][90909090](03)    8b4df8            mov ecx,[ebp-08]
...[00000bf9][001017aa][00000000](01)    51                push ecx
...[00000bfa][001017a6][00000343](05)    6843030000        push 00000343
---[00000bff][001017a6][00000343](05)    e8d4f7ffff        call 000003d8
Input_Would_Halt2 = 0
...[00000c04][001017ae][90909090](03)    83c408            add esp,+08
...[00000c07][001017aa][00000003](02)    6a03              push +03
...[00000c09][001017a6][00000b18](05)    68180b0000        push 00000b18
...[00000c0e][001017a2][00000c13](05)    e8a5fdffff        call 000009b8
Begin Local Halt Decider Simulation at Machine Address:b18
...[00000b18][002a6cb6][002a6cba](01)    55                push ebp
...[00000b19][002a6cb6][002a6cba](02)    8bec              mov ebp,esp
...[00000b1b][002a6cb2][00296c86](01)    51                push ecx
...[00000b1c][002a6cb2][00296c86](03)    8b4508            mov eax,[ebp+08]
...[00000b1f][002a6cae][00000003](01)    50                push eax
...[00000b20][002a6caa][00000b25](05)    e8f3ffffff        call 00000b18
...[00000b18][002a6ca6][002a6cb6](01)    55                push ebp
...[00000b19][002a6ca6][002a6cb6](02)    8bec              mov ebp,esp
...[00000b1b][002a6ca2][00296c86](01)    51                push ecx
```

```
...[00000b1c][002a6ca2][00296c86](03)    8b4508          mov eax,[ebp+08]
...[00000b1f][002a6c9e][00000003](01)    50              push eax
...[00000b20][002a6c9a][00000b25](05)    e8f3ffffff      call 00000b18
Local Halt Decider: Infinite Recursion Detected Simulation Stopped
...[00000c13][001017ae][90909090](03)    83c408          add esp,+08
...[00000c16][001017ae][90909090](03)    8945f4          mov [ebp-0c],eax
...[00000c19][001017ae][90909090](03)    8b55f4          mov edx,[ebp-0c]
...[00000c1c][001017aa][00000000](01)    52              push edx
...[00000c1d][001017a6][0000035b](05)    685b030000      push 0000035b
---[00000c22][001017a6][0000035b](05)    e8b1f7ffff      call 000003d8
Input_Would_Halt3 = 0
...[00000c27][001017ae][90909090](03)    83c408          add esp,+08
...[00000c2a][001017aa][00000b58](05)    68580b0000      push 00000b58
...[00000c2f][001017a6][00000b58](05)    68580b0000      push 00000b58
...[00000c34][001017a2][00000c39](05)    e87ffdffff      call 000009b8
Begin Local Halt Decider Simulation at Machine Address:b58
...[00000b58][002f16de][002f16e2](01)    55              push ebp
...[00000b59][002f16de][002f16e2](02)    8bec            mov ebp,esp
...[00000b5b][002f16da][002e16ae](01)    51              push ecx
...[00000b5c][002f16da][002e16ae](03)    8b4508          mov eax,[ebp+08]
...[00000b5f][002f16d6][00000b58](01)    50              push eax
...[00000b60][002f16d6][00000b58](03)    8b4d08          mov ecx,[ebp+08]
...[00000b63][002f16d2][00000b58](01)    51              push ecx
...[00000b64][002f16ce][00000b69](05)    e8cfffffff      call 00000b38
...[00000b38][002f16ca][002f16de](01)    55              push ebp
...[00000b39][002f16ca][002f16de](02)    8bec            mov ebp,esp
...[00000b3b][002f16ca][002f16de](03)    8b450c          mov eax,[ebp+0c]
...[00000b3e][002f16c6][00000b58](01)    50              push eax
Calling:_H_Hat2()
Decode_Control_Flow_Instruction([00000008][002f16ca][00000b58])
...[00000b3f][002f16c2][00000b42](03)    ff5508          call dword [ebp+08]
Decode_Control_Flow_Instruction([00000008][0010174e][0010179a])
...[00000b58][002f16be][002f16ca](01)    55              push ebp
...[00000b59][002f16be][002f16ca](02)    8bec            mov ebp,esp
...[00000b5b][002f16ba][00000b58](01)    51              push ecx
...[00000b5c][002f16ba][00000b58](03)    8b4508          mov eax,[ebp+08]
...[00000b5f][002f16b6][00000b58](01)    50              push eax
...[00000b60][002f16b6][00000b58](03)    8b4d08          mov ecx,[ebp+08]
...[00000b63][002f16b2][00000b58](01)    51              push ecx
...[00000b64][002f16ae][00000b69](05)    e8cfffffff      call 00000b38
Local Halt Decider: Infinite Recursion Detected Simulation Stopped
...[00000c39][001017ae][90909090](03)    83c408          add esp,+08
...[00000c3c][001017ae][90909090](03)    8945f0          mov [ebp-10],eax
...[00000c3f][001017ae][90909090](03)    8b45f0          mov eax,[ebp-10]
...[00000c42][001017aa][00000000](01)    50              push eax
...[00000c43][001017a6][00000373](05)    6873030000      push 00000373
---[00000c48][001017a6][00000373](05)    e88bf7ffff      call 000003d8
Input_Would_Halt4 = 0
...[00000c4d][001017ae][90909090](03)    83c408          add esp,+08
...[00000c50][001017aa][00000b88](05)    68880b0000      push 00000b88
...[00000c55][001017a6][00000b88](05)    68880b0000      push 00000b88
...[00000c5a][001017a2][00000c5f](05)    e8d9feffff      call 00000b38
...[00000b38][0010179e][001017c6](01)    55              push ebp
...[00000b39][0010179e][001017c6](02)    8bec            mov ebp,esp
...[00000b3b][0010179e][001017c6](03)    8b450c          mov eax,[ebp+0c]
...[00000b3e][0010179a][00000b88](01)    50              push eax
Calling:_H_Hat()
...[00000b3f][00101796][00000b42](03)    ff5508          call dword [ebp+08]
Decode_Control_Flow_Instruction([00000008][0010179e][00000b88])
...[00000b88][00101792][0010179e](01)    55              push ebp
...[00000b89][00101792][0010179e](02)    8bec            mov ebp,esp
...[00000b8b][0010178e][000009c7](01)    51              push ecx
...[00000b8c][0010178e][000009c7](03)    8b4508          mov eax,[ebp+08]
...[00000b8f][0010178a][00000b88](01)    50              push eax
...[00000b90][0010178a][00000b88](03)    8b4d08          mov ecx,[ebp+08]
...[00000b93][00101786][00000b88](01)    51              push ecx
```

```
...[00000b94][00101782][00000b99](05)    e81ffeffff        call 000009b8
Begin Local Halt Decider Simulation at Machine Address:b88
...[00000b88][0033c106][0033c10a](01)    55                push ebp
...[00000b89][0033c106][0033c10a](02)    8bec              mov ebp,esp
...[00000b8b][0033c102][0032c0d6](01)    51                push ecx
...[00000b8c][0033c102][0032c0d6](03)    8b4508            mov eax,[ebp+08]
...[00000b8f][0033c0fe][00000b88](01)    50                push eax
...[00000b90][0033c0fe][00000b88](03)    8b4d08            mov ecx,[ebp+08]
...[00000b93][0033c0fa][00000b88](01)    51                push ecx
...[00000b94][0033c0f6][00000b99](05)    e81ffeffff        call 000009b8
...[00000b88][00386b2e][00386b32](01)    55                push ebp
...[00000b89][00386b2e][00386b32](02)    8bec              mov ebp,esp
...[00000b8b][00386b2a][00376afe](01)    51                push ecx
...[00000b8c][00386b2a][00376afe](03)    8b4508            mov eax,[ebp+08]
...[00000b8f][00386b26][00000b88](01)    50                push eax
...[00000b90][00386b26][00000b88](03)    8b4d08            mov ecx,[ebp+08]
...[00000b93][00386b22][00000b88](01)    51                push ecx
...[00000b94][00386b1e][00000b99](05)    e81ffeffff        call 000009b8
Local Halt Decider: Infinite Recursion Detected Simulation Stopped
...[00000b99][0010178e][000009c7](03)    83c408            add esp,+08
...[00000b9c][0010178e][00000000](03)    8945fc            mov [ebp-04],eax
...[00000b9f][0010178e][00000000](04)    837dfc00          cmp dword [ebp-04],+00
...[00000ba3][0010178e][00000000](02)    7402              jz 00000ba7
...[00000ba7][00101792][0010179e](02)    8be5              mov esp,ebp
...[00000ba9][00101796][00000b42](01)    5d                pop ebp
...[00000baa][0010179a][00000b88](01)    c3                ret
...[00000b42][0010179e][001017c6](03)    83c404            add esp,+04
...[00000b45][0010179e][001017c6](05)    b801000000        mov eax,00000001
...[00000b4a][001017a2][00000c5f](01)    5d                pop ebp
...[00000b4b][001017a6][00000b88](01)    c3                ret
...[00000c5f][001017ae][90909090](03)    83c408            add esp,+08
...[00000c62][001017ae][90909090](03)    8945ec            mov [ebp-14],eax
...[00000c65][001017ae][90909090](03)    8b4dec            mov ecx,[ebp-14]
...[00000c68][001017aa][00000001](01)    51                push ecx
...[00000c69][001017a6][0000038b](05)    688b030000        push 0000038b
---[00000c6e][001017a6][0000038b](05)    e865f7ffff        call 000003d8
Input_Would_Halt5 = 1
...[00000c73][001017ae][90909090](03)    83c408            add esp,+08
...[00000c76][001017aa][00000b88](05)    68880b0000        push 00000b88
...[00000c7b][001017a6][00000b88](05)    68880b0000        push 00000b88
...[00000c80][001017a2][00000c85](05)    e833fdffff        call 000009b8
Begin Local Halt Decider Simulation at Machine Address:b88
...[00000b88][00396bce][00396bd2](01)    55                push ebp
...[00000b89][00396bce][00396bd2](02)    8bec              mov ebp,esp
...[00000b8b][00396bca][00386b9e](01)    51                push ecx
...[00000b8c][00396bca][00386b9e](03)    8b4508            mov eax,[ebp+08]
...[00000b8f][00396bc6][00000b88](01)    50                push eax
...[00000b90][00396bc6][00000b88](03)    8b4d08            mov ecx,[ebp+08]
...[00000b93][00396bc2][00000b88](01)    51                push ecx
...[00000b94][00396bbe][00000b99](05)    e81ffeffff        call 000009b8
...[00000b88][003e15f6][003e15fa](01)    55                push ebp
...[00000b89][003e15f6][003e15fa](02)    8bec              mov ebp,esp
...[00000b8b][003e15f2][003d15c6](01)    51                push ecx
...[00000b8c][003e15f2][003d15c6](03)    8b4508            mov eax,[ebp+08]
...[00000b8f][003e15ee][00000b88](01)    50                push eax
...[00000b90][003e15ee][00000b88](03)    8b4d08            mov ecx,[ebp+08]
...[00000b93][003e15ea][00000b88](01)    51                push ecx
...[00000b94][003e15e6][00000b99](05)    e81ffeffff        call 000009b8
Local Halt Decider: Infinite Recursion Detected Simulation Stopped
...[00000c85][001017ae][90909090](03)    83c408            add esp,+08
...[00000c88][001017ae][00000000](03)    8945e8            mov [ebp-18],eax
...[00000c8b][001017ae][00000000](03)    8b55e8            mov edx,[ebp-18]
...[00000c8e][001017aa][00000000](01)    52                push edx
...[00000c8f][001017a6][000003a3](05)    68a3030000        push 000003a3
---[00000c94][001017a6][000003a3](05)    e83ff7ffff        call 000003d8
Input_Would_Halt6 = 0
```

```
...[00000c99][001017ae][00000000](03)    83c408         add esp,+08
...[00000c9c][001017aa][00000b88](05)    68880b0000     push 00000b88
...[00000ca1][001017a6][00000ca6](05)    e8e2feffff     call 00000b88
...[00000b88][001017a2][001017c6](01)    55             push ebp
...[00000b89][001017a2][001017c6](02)    8bec           mov ebp,esp
...[00000b8b][0010179e][000009c7](01)    51             push ecx
...[00000b8c][0010179e][000009c7](03)    8b4508         mov eax,[ebp+08]
...[00000b8f][0010179a][00000b88](01)    50             push eax
...[00000b90][0010179a][00000b88](03)    8b4d08         mov ecx,[ebp+08]
...[00000b93][00101796][00000b88](01)    51             push ecx
...[00000b94][00101792][00000b99](05)    e81ffeffff     call 000009b8
Begin Local Halt Decider Simulation at Machine Address:b88
...[00000b88][003f1696][003f169a](01)    55             push ebp
...[00000b89][003f1696][003f169a](02)    8bec           mov ebp,esp
...[00000b8b][003f1692][003e1666](01)    51             push ecx
...[00000b8c][003f1692][003e1666](03)    8b4508         mov eax,[ebp+08]
...[00000b8f][003f168e][00000b88](01)    50             push eax
...[00000b90][003f168e][00000b88](03)    8b4d08         mov ecx,[ebp+08]
...[00000b93][003f168a][00000b88](01)    51             push ecx
...[00000b94][003f1686][00000b99](05)    e81ffeffff     call 000009b8
...[00000b88][0043c0be][0043c0c2](01)    55             push ebp
...[00000b89][0043c0be][0043c0c2](02)    8bec           mov ebp,esp
...[00000b8b][0043c0ba][0042c08e](01)    51             push ecx
...[00000b8c][0043c0ba][0042c08e](03)    8b4508         mov eax,[ebp+08]
...[00000b8f][0043c0b6][00000b88](01)    50             push eax
...[00000b90][0043c0b6][00000b88](03)    8b4d08         mov ecx,[ebp+08]
...[00000b93][0043c0b2][00000b88](01)    51             push ecx
...[00000b94][0043c0ae][00000b99](05)    e81ffeffff     call 000009b8
Local Halt Decider: Infinite Recursion Detected Simulation Stopped
...[00000b99][0010179e][000009c7](03)    83c408         add esp,+08
...[00000b9c][0010179e][00000000](03)    8945fc         mov [ebp-04],eax
...[00000b9f][0010179e][00000000](04)    837dfc00       cmp dword [ebp-04],+00
...[00000ba3][0010179e][00000000](02)    7402           jz 00000ba7
...[00000ba7][001017a2][001017c6](02)    8be5           mov esp,ebp
...[00000ba9][001017a6][00000ca6](01)    5d             pop ebp
...[00000baa][001017aa][00000b88](01)    c3             ret
...[00000ca6][001017ae][00000000](03)    83c404         add esp,+04
...[00000ca9][001017ae][00000000](02)    33c0           xor eax,eax
...[00000cab][001017c6][00000000](02)    8be5           mov esp,ebp
...[00000cad][001017ca][00100000](01)    5d             pop ebp
...[00000cae][001017ce][0000039c](01)    c3             ret
Number_of_User_Instructions(232)
Number of Instructions Executed(87815)
```