```
In[37]:= p = FromDigits["FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFEBAAEDCE6AF48A03BBFD25E8CD0364141", 16];

In[38]:= r0 = Mod[2^256 - p, 2^52];

In[39]:= r1 = Mod[Floor[(2^256 - p) / 2^52], 2^52];

In[40]:= r2 = Mod[Floor[(2^256 - p) / 2^104], 2^52];

In[41]:= aLimbs = {1 981 404 003 755 835, 1 592 544 783 698 284,
            4 150 950 532 038 206, 3 135 904 464 385 237, 77 854 302 420 060};
         bLimbs = {1 311 085 228 448 312, 3 942 750 054 222 886, 2 120 479 608 659 615,
            3 780 699 925 647 021, 223 709 799 478 902};

In[43]:= a = Plus @@ Times @@@ Transpose@{aLimbs, Table[2^(52 i), {i, 0, 4}]};

In[44]:= b = Plus @@ Times @@@ Transpose@{bLimbs, Table[2^(52 i), {i, 0, 4}]};

In[45]:= result = Mod[a b, p];

In[46]:= ai[n_] := (Reverse@IntegerDigits[a, 2^52])[[n + 1]]

In[47]:= bi[n_] := (Reverse@IntegerDigits[b, 2^52])[[n + 1]]

In[48]:= ai[n_] := Mod[aLimbs[[n + 1]], 2^52]

In[49]:= bi[n_] := Mod[bLimbs[[n + 1]], 2^52]

In[50]:= cl[n_] := Mod[Sum[ai[i] bi[n - i], {i, Max[0, n - 4], Min[n, 4]}], 2^52]

In[51]:= cu[n_] := Floor[Sum[ai[i] bi[n - i], {i, Max[0, n - 4], Min[n, 4]}] / 2^52]
```

0

```
In[52]:= res0 = (cl[0] + cu[0] 2^52) + 2^52 (cl[1] + cu[1] 2^52) + 2^104 (cl[2] + cu[2] 2^52) +
            2^156 (cl[3] + cu[3] 2^52) + 2^208 (cl[4] + cu[4] 2^52) + 2^260 (cl[5] + cu[5] 2^52) +
            2^312 (cl[6] + cu[6] 2^52) + 2^364 (cl[7] + cu[7] 2^52) + 2^416 (cl[8] + cu[8] 2^52);

In[53]:= Mod[res0, p] == result

Out[53]= True
```

1

```
In[54]:= res1 = (cl[0] + 16 r0 (cu[4] + cl[5]) + 256 r0 (r1 cu[8] + r2 (cu[7] + cl[8]))) +
            2^52 (cu[0] + cl[1] + 16 r0 (cu[5] + cl[6]) + 16 r1 (cu[4] + cl[5]) + 256 r0 r2 cu[8] + 256 r1 r1 cu[8] +
               256 r1 r2 (cu[7] + cl[8])) + 2^104 (cu[1] + cl[2] + 16 r0 (cu[6] + cl[7]) + 16 r1 (cu[5] + cl[6]) +
               16 r2 (cu[4] + cl[5]) + 256 r1 r2 cu[8] + 256 r2 r1 cu[8] + 256 r2 r2 (cu[7] + cl[8])) +
            2^156 (cu[2] + cl[3] + 16 r0 (cu[7] + cl[8]) + 16 r1 (cu[6] + cl[7]) + 16 r2 (cu[5] + cl[6]) + 256 r2 r2 cu[8]) +
            2^208 (cu[3] + cl[4] + 16 r0 cu[8] + 16 r1 (cu[7] + cl[8]) + 16 r2 (cu[6] + cl[7]));
```

```
In[55]:=  Mod[res1, p] == result

Out[55]=  True
```

2

```
In[56]:=  s01 = r0 r1;
          s01l = Mod[s01, 2^52];
          s01u = Mod[Floor[s01/2^52], 2^52];

In[59]:=  s02 = r0 r2;
          s02l = Mod[s02, 2^52];
          s02u = Mod[Floor[s02/2^52], 2^52];

In[62]:=  s11 = r1 r1;
          s11l = Mod[s11, 2^52];
          s11u = Mod[Floor[s11/2^52], 2^52];

In[65]:=  s12 = r1 r2;
          s12l = Mod[s12, 2^52];
          s12u = Mod[Floor[s12/2^52], 2^52];

In[68]:=  s22 = r2 r2;
          s22l = Mod[s22, 2^52];
          s22u = Mod[Floor[s22/2^52], 2^52];

In[71]:=  f0 = r0 cu[8];
          f0l = Mod[f0, 2^52];
          f0u = Mod[Floor[f0/2^52], 2^52];

In[74]:=  f1 = r1 (cu[7] + cl[8]);
          f1l = Mod[f1, 2^52];
          f1u = Mod[Floor[f1/2^52], 2^52];

In[77]:=  f2 = r2 (cu[6] + cl[7]);
          f2l = Mod[f2, 2^52];
          f2u = Mod[Floor[f2/2^52], 2^52];

In[80]:=  f3 = s22u cu[8];
          f3l = Mod[f3, 2^52];
          f3u = Mod[Floor[f3/2^52], 2^52];
```

```
In[83]:= res2 = (cl[0] + 16 r0 (cu[4] + cl[5]) + 256 s01l cu[8] + 256 s02l (cu[7] + cl[8]) + 256 r0 (f0u + f1u + f2u + 16 f3u)) +
           2 ^ 52 (cu[0] + cl[1] + 16 r0 (cu[5] + cl[6]) + 16 r1 (cu[4] + cl[5]) + 256 s02l cu[8] + 256 s11l cu[8] +
              256 s12l (cu[7] + cl[8]) + 256 s01u cu[8] + 256 s02u (cu[7] + cl[8]) + 256 r1 (f0u + f1u + f2u + 16 f3u)) +
           2 ^ 104 (cu[1] + cl[2] + 16 r0 (cu[6] + cl[7]) + 16 r1 (cu[5] + cl[6]) + 16 r2 (cu[4] + cl[5]) +
              512 s12l cu[8] + 256 s22l (cu[7] + cl[8]) + 256 s02u cu[8] + 256 s11u cu[8] +
              256 s12u (cu[7] + cl[8]) + 256 r2 (f0u + f1u + f2u + 16 f3u)) +
           2 ^ 156 (cu[2] + cl[3] + 16 r0 (cu[7] + cl[8]) + 16 r1 (cu[6] + cl[7]) + 16 r2 (cu[5] + cl[6]) +
              256 s22l cu[8] + 256 s12u cu[8] + 256 s12u cu[8] + 256 s22u (cu[7] + cl[8])) +
           2 ^ 208 (cu[3] + cl[4] + 16 (f0l + f1l + f2l + 16 f3l));

In[84]:= Mod[res2, p] == result

Out[84]= True
```

Debugging

```
In[85]:= splitMul[x_, y_] := IntegerDigits[x * y, 2 ^ 52]

In[86]:= join[x_, y_] := x + y 2 ^ 52

In[87]:= d1 = {16 r0 (cu[4] + cl[5]), 256 s01l cu[8], 256 s02l (cu[7] + cl[8]), 256 r0 (f0u + f1u + f2u + 16 f3u)};

In[88]:= d2 = {16 r0 (cu[5] + cl[6]), 16 r1 (cu[4] + cl[5]), 256 s02l cu[8], 256 s11l cu[8],
           256 s12l (cu[7] + cl[8]), 256 s01u cu[8], 256 s02u (cu[7] + cl[8]), 256 r1 (f0u + f1u + f2u + 16 f3u)};

In[89]:= d3 = {16 r0 (cu[6] + cl[7]), 16 r1 (cu[5] + cl[6]), 16 r2 (cu[4] + cl[5]), 512 s12l cu[8], 256 s22l (cu[7] + cl[8]),
           256 s02u cu[8], 256 s11u cu[8], 256 s12u (cu[7] + cl[8]), 256 r2 (f0u + f1u + f2u + 16 f3u)};

In[90]:= d4 = {16 r0 (cu[7] + cl[8]), 16 r1 (cu[6] + cl[7]), 16 r2 (cu[5] + cl[6]),
           256 s22l cu[8], 256 s12u cu[8], 256 s12u cu[8], 256 s22u (cu[7] + cl[8])};

In[91]:= cl[0] + Plus @@ (Mod[#, 2 ^ 52] & /@ d1) == 14 219 843 304 969 114

Out[91]= False

In[92]:= cu[0] + cl[1] + Plus @@ (Mod[#, 2 ^ 52] & /@ d2) + Plus @@ (Floor[# / 2 ^ 52] & /@ d1) == 1 087 833 768 451 024 963

Out[92]= False

In[93]:= cu[1] + cl[2] + Plus @@ (Mod[#, 2 ^ 52] & /@ d3) + Plus @@ (Floor[# / 2 ^ 52] & /@ d2) == 959 867 073 997 061 848

Out[93]= False

In[94]:= cu[2] + cl[3] + Plus @@ (Mod[#, 2 ^ 52] & /@ d4) + Plus @@ (Floor[# / 2 ^ 52] & /@ d3) == 185 606 254 470 553 038

Out[94]= False

In[95]:= cu[3] + cl[4] + 16 (f0l + f1l + f2l + 16 f3l) + Plus @@ (Floor[# / 2 ^ 52] & /@ d4) == 166 075 137 468 553 819

Out[95]= False

In[96]:= ansLimbs = {16 593 146 405 017 560, 936 547 462 867 448 409,
           806 495 106 354 285 297, 215 451 228 430 912 425, 143 009 125 421 609 081};
```

```
In[97]:=  ans = Plus @@ Times @@@ Transpose@{ansLimbs, Table[2^(52 i), {i, 0, 4}]};

In[98]:=  Mod[ans, p]

Out[98]=  8 160 572 387 813 461 671 324 022 432 516 612 737 382 221 156 192 548 259 081 941 912 922 232 183 516

In[99]:=  result

Out[99]=  8 160 572 387 813 461 671 324 022 432 516 612 737 382 221 156 192 548 259 081 941 912 922 232 183 516

In[100]:=  a

Out[100]=  32 027 402 359 818 319 085 220 203 671 367 342 494 338 542 732 862 813 126 078 940 774 148 405 520 187

In[101]:=  b

Out[101]=  92 028 873 639 986 946 753 274 199 041 849 688 314 715 603 778 086 907 388 764 316 572 332 718 021 176

In[102]:=  BaseForm[s01l, 16]
```

Out[102]//BaseForm=

$777920542397e_{16}$

```
In[103]:=  BaseForm[s01u, 16]
```

Out[103]//BaseForm=

$15910772c569a_{16}$

```
In[104]:=  BaseForm[s02l, 16]
```

Out[104]//BaseForm=

$e2ffd866a831d_{16}$

```
In[105]:=  BaseForm[s02u, 16]
```

Out[105]//BaseForm=

1152492_{16}

```
In[106]:=  BaseForm[s11l, 16]
```

Out[106]//BaseForm=

$cbaebca011004_{16}$

```
In[107]:=  BaseForm[s11u, 16]
```

Out[107]//BaseForm=

$280dd43d3893_{16}$

```
In[108]:=  BaseForm[s12l, 16]
```

Out[108]//BaseForm=

$cca28498bee46_{16}$

```
In[109]:=  BaseForm[s12u, 16]
```

Out[109]//BaseForm=

$202b7e_{16}$

```
In[110]:=  BaseForm[s22l, 16]
```

Out[110]//BaseForm=

$19d671c952ac9_{16}$

```
In[111]:=  BaseForm[FromDigits["1000000000", 16] - FromDigits["EFFFFFC2F", 16], 16]
Out[111]//BaseForm=
           1000003d1₁₆

In[112]:=  BaseForm[s12u, 16]
Out[112]//BaseForm=
           202b7e₁₆

In[113]:=  N@Log2[s22l]
Out[113]=  48.6914
```

ASM Debugging

```
In[216]:=  c0 = ai[0] bi[0];
           c1 = ai[0] bi[1] + ai[1] bi[0];
           c2 = ai[0] bi[2] + ai[1] bi[1] + ai[2] bi[0];
           c3 = ai[0] bi[3] + ai[1] bi[2] + ai[2] bi[1] + ai[3] bi[0];
           c4 = ai[0] bi[4] + ai[1] bi[3] + ai[2] bi[2] + ai[3] bi[1] + ai[4] bi[0];
           c5 = ai[1] bi[4] + ai[2] bi[3] + ai[3] bi[2] + ai[4] bi[1];
           c6 = ai[2] bi[4] + ai[3] bi[3] + ai[4] bi[2];
           c7 = ai[3] bi[4] + ai[4] bi[3];
           c8 = ai[4] bi[4];
           l0 = c0 + (2 ^ 4) r0 c5 + (2 ^ 8) r0 r2 c8;
           l1 = c1 + (2 ^ 4) r1 c5 + (2 ^ 4) r0 c6 + (2 ^ 8) r1 r2 c8;
           l2 = c2 + (2 ^ 4) r2 c5 + (2 ^ 4) r1 c6 + (2 ^ 4) r0 c7 + (2 ^ 8) (r2 ^ 2) c8;
           l3 = c3 + (2 ^ 4) r2 c6 + (2 ^ 4) r1 c7 + (2 ^ 4) r0 c8;
           l4 = c4 + (2 ^ 4) r2 c7 + (2 ^ 4) r1 c8;
           num = Mod[Plus @@ Times @@@ Transpose@{{l0, l1, l2, l3, l4}, Table[2 ^ (52 i), {i, 0, 4}]}, p];

In[231]:=  num == result
Out[231]=  True
```

```
In[239]:= l0 = c0 + (2 ^ 4) r0 cl[5] + (2 ^ 8) s02l cl[8] + ((2 ^ 8) s02l cu[7] + (2 ^ 8) s01l cu[8]);
          l1 =
            c1 + (2 ^ 4) r1 cl[5] + (2 ^ 4) r0 cl[6] + (2 ^ 8) s12l cl[8] + ((2 ^ 4) r0 cu[5] + (2 ^ 8) s02l cu[8] + (2 ^ 8) s02u cl[8]) +
              ((2 ^ 8) s12l cu[7] + (2 ^ 8) s11l cu[8] + (2 ^ 8) s02u cu[7] + (2 ^ 8) s01u cu[8]);
          l2 = c2 + (2 ^ 4) r2 cl[5] + (2 ^ 4) r1 cl[6] + (2 ^ 4) r0 cl[7] + (2 ^ 8) s22l cl[8] +
              ((2 ^ 4) r1 cu[5] + (2 ^ 4) r0 cu[6] + (2 ^ 8) s12l cu[8] + (2 ^ 8) s12u cl[8]) + ((2 ^ 8) s02u cu[8]) +
              ((2 ^ 8) s22l cu[7] + (2 ^ 8) s12l cu[8] + (2 ^ 8) s12u cu[7] + (2 ^ 8) s11u cu[8]);
          l3 = c3 + (2 ^ 4) r2 cl[6] + (2 ^ 4) r1 cl[7] + (2 ^ 4) r0 cl[8] +
              ((2 ^ 4) r2 cu[5] + (2 ^ 4) r1 cu[6] + (2 ^ 4) r0 cu[7] + (2 ^ 8) s22l cu[8]) + ((2 ^ 8) s12u cu[8]) + ((2 ^ 8) s12u cu[8]);
          l4 = c4 + (2 ^ 4) r2 cl[7] + (2 ^ 4) r1 cl[8] + ((2 ^ 4) r2 cu[6] + (2 ^ 4) r1 cu[7] + (2 ^ 4) r0 cu[8]);
          num = Mod[Plus @@ Times @@@ Transpose@{{l0, l1, l2, l3, l4}, Table[2 ^ (52 i), {i, 0, 4}]}, p];

In[245]:= num == result

Out[245]= True
```

```
In[420]:= d = (2^9) s12u (BitShiftRight[c8, 52]) +
              (2^4) r2 (BitShiftRight[c5, 52]) + (2^4) r1 (BitShiftRight[c6, 52]) +
              (2^4) r0 (BitShiftRight[c7, 52]) + (2^8) s22l (BitShiftRight[c8, 52]);
          d += ai[0] bi[3] + ai[1] bi[2] + ai[2] bi[1] + ai[3] bi[0] + (2^4) r2 (BitAnd[c6, M]) +
              (2^4) r1 (BitAnd[c7, M]) + (2^4) r0 (BitAnd[c8, M]);
          asmr3 = BitAnd[d, M];
          d = BitShiftRight[d, 52];
          d +=
            (2^4) r2 (BitShiftRight[c6, 52]) + (2^4) r1 (BitShiftRight[c7, 52]) + (2^4) r0 (BitShiftRight[c8, 52]);
          d += ai[0] bi[4] + ai[1] bi[3] + ai[2] bi[2] + ai[3] bi[1] + ai[4] bi[0] +
              (2^4) r2 (BitAnd[c7, M]) + (2^4) r1 (BitAnd[c8, M]);
          asmr4 = BitAnd[d, M];
          d = BitShiftRight[d, 52];
          tmp1 = d;
          d = tmp1 (2^4) r0;
          d += (2^8) s02l (BitShiftRight[c7, 52]) + (2^8) s01l (BitShiftRight[c8, 52]);
          d += ai[0] bi[0] + (2^4) r0 (BitAnd[c5, M]) + (2^8) s02l (BitAnd[c8, M]);
          asmr0 = BitAnd[d, M];
          d = BitShiftRight[d, 52];
          d += tmp1 (2^4) r1;
          d += (2^8) s02u (BitShiftRight[c7, 52]) + (2^8) s01u (BitShiftRight[c8, 52]) +
              (2^8) s12l (BitShiftRight[c7, 52]) + (2^8) s11l (BitShiftRight[c8, 52]);
          d += (2^4) r0 (BitShiftRight[c5, 52]) + (2^8) s02l (BitShiftRight[c8, 52]) + (2^8) s02u (BitAnd[c8, M]);
          d += ai[0] bi[1] + ai[1] bi[0] + (2^4) r1 (BitAnd[c5, M]) + (2^4) r0 (BitAnd[c6, M]) + (2^8) s12l (BitAnd[c8, M]);
          asmr1 = BitAnd[d, M];
          d = BitShiftRight[d, 52];
          d += tmp1 (2^4) r2;
          d += (2^8) s12u (BitShiftRight[c7, 52]) + (2^8) s11u (BitShiftRight[c8, 52]) +
              (2^8) s22l (BitShiftRight[c7, 52]) + (2^8) s12l (BitShiftRight[c8, 52]);
          d += (2^8) s02u cu[8];
          d += (2^4) r1 (BitShiftRight[c5, 52]) + (2^4) r0 (BitShiftRight[c6, 52]) +
              (2^8) s12l (BitShiftRight[c8, 52]) + (2^8) s12u (BitAnd[c8, M]);
          d += ai[0] bi[2] + ai[1] bi[1] + ai[2] bi[0] + (2^4) r2 (BitAnd[c5, M]) +
              (2^4) r1 (BitAnd[c6, M]) + (2^4) r0 (BitAnd[c7, M]) + (2^8) s22l (BitAnd[c8, M]);
          asmr2 = BitAnd[d, M];
          d = BitShiftRight[d, 52];
          asmr3 += BitAnd[d, M];
          d = BitShiftRight[d, 52];
          asmr4 += d;
```

```
In[451]:=  Mod[Plus @@ Times @@@ Transpose@{{asmr0, asmr1, asmr2, asmr3, asmr4}, Table[2^(52 i), {i, 0, 4}]},
             p] == result

Out[451]=  True

In[453]:=  BaseForm[s12u * 2, 16]
```

Out[453]//BaseForm=

$4056fc_{16}$

ASM

```
In[125]:=  M = (2^52) - 1;

In[126]:=  carry = 0;
           rax = 0;
           rdx = 0;
           rcx = 0;
           r15 = 0;
           r8 = 0;
           r9 = 0;
           asmr0 = 0;
           asmr1 = 0;
           asmr2 = 0;
           asmr3 = 0;
           asmr4 = 0;

In[138]:=  addq[x_, y_] := Block[{sum},
           sum = x + y;
           carry = Floor[sum / (2^64)];
           Mod[sum, 2^64]
           ]

In[139]:=  adcq[x_, y_] := Block[{sum},
           sum = x + y + carry;
           carry = Floor[sum / (2^64)];
           Mod[sum, 2^64]
           ]

In[140]:=  mulq[x_] := Block[{prod},
           prod = rax * x;
           {rdx, rax} = IntegerDigits[prod, 2^64];
           ]
```

```
In[141]:= shrdq[s_, x_, y_] := Block[{n},
          n = y + (2 ^ 64) x;
          Mod[BitShiftRight[n, s], 2 ^ 64]
          ]
```

r[3] partial

```
In[142]:= rax = ai[0];
          mulq[bi[3]];
          {rcx, r15} = {rax, rdx};
          rax = ai[1];
          mulq[bi[2]];
          {rcx, r15} += {rax, rdx};
          rax = ai[2];
          mulq[bi[1]];
          {rcx, r15} += {rax, rdx};
          rax = ai[3];
          mulq[bi[0]];
          {rcx, r15} += {rax, rdx};
          rax = BitAnd[c6, M];
          mulq[BitShiftLeft[r2, 4]];
          {rcx, r15} += {rax, rdx};
          rax = BitAnd[c7, M];
          mulq[BitShiftLeft[r1, 4]];
          {rcx, r15} += {rax, rdx};
          asmr3 = BitAnd[rax, M];
```

r[4] partial

```
In[161]:= rcx = shrdq[52, r15, rcx];
          r15 = 0;
          {r8, r9} = Reverse@IntegerDigits[c6, 2 ^ 64];
          r8 = shrdq[52, r9, r8];
          rax = r8;
          mulq[BitShiftLeft[r2, 4]];
          {rcx, r15} += {rax, rdx};
```