```
         |                        /     Michael Beaver
         |                        /     CS 311 - Fall 2013
         |                        /
         |                        /     This program raises an integer to a given power. The first input
         |                        /     is the base, and the second input is the exponent. The result is
         |                        /     stored in Temp, which is printed to the output region.
         |                        /
         |                        /
         |          ORG 100
100 5000 |  Start   INPUT
101 2141 |          STORE Base      /Input Base b
102 2144 |          STORE Temp
103 2145 |          STORE Temp2
104 8000 |          SKIPCOND 000    /Skip if negative Base
105 9107 |          JUMP GetExp
106 016C |          JNS GetAbs      /Calculate |b| (abs. value)
         |                          /
107 5000 |  GetExp INPUT            /Input Exponent n
108 2142 |          STORE Expnt
         |                          /
109 1141 |          LOAD Base
10A 8400 |          SKIPCOND 400    /Checking for 0^n
10B 910D |          JUMP Here       /Not 0^n
10C 913B |          JUMP EOne       /Go to output 0
         |                          /
10D 1142 |  Here    LOAD Expnt
10E 8800 |          SKIPCOND 800    /Skip if positive Exponent
10F 9138 |          JUMP EZero      /b^0 => output 1
         |                          /
110 1141 |          LOAD Base       /Testing if Base == 1
111 414A |          SUBT One
112 8800 |          SKIPCOND 800
113 9115 |          JUMP Btween     /Base <= 1
114 9118 |          JUMP There      /Base > 1
115 8400 |  Btween SKIPCOND 400     /Skip if Base == 1
116 8000 |          SKIPCOND 000    /Skip if Base != 1
117 9138 |          JUMP EZero      /Base == 1
         |                          /
118 1142 |  There   LOAD Expnt
119 414A |          SUBT One
11A 8800 |          SKIPCOND 800    /Checking if exponent > 1
11B 913B |          JUMP EOne       /b^1 => output b
         |                          /
11C 2148 |          STORE Ctr1
11D 1141 |          LOAD Base
11E 8800 |          SKIPCOND 800    /Checking for base's sign
```

```
11F 9127 |          JUMP NegBs       /Negative Base
         |                           /
120 014C |  ELoop1 JNS Mult          /Exponentiate positive base
121 1148 |          LOAD Ctr1
122 414A |          SUBT One
123 2148 |          STORE Ctr1
124 8400 |          SKIPCOND 400
125 9120 |          JUMP ELoop1
126 913E |          JUMP End
         |                           /
127 1141 |  NegBs  LOAD Base
128 2146 |          STORE Temp3
129 2144 |          STORE Temp
12A 015B |  ELoop2 JNS MulNeg        /Exponentiate negative base
12B 1148 |          LOAD Ctr1
12C 414A |          SUBT One
12D 2148 |          STORE Ctr1
12E 8400 |          SKIPCOND 400
12F 912A |          JUMP ELoop2
130 1146 |          LOAD Temp3
131 2144 |          STORE Temp
         |                           /
132 017C |          JNS Div2         /Testing for even exponent
133 8000 |          SKIPCOND 000
134 9136 |          JUMP EvnExp
135 913E |  OddExp JUMP End          /Do nothing if odd exponent
136 016C |  EvnExp JNS GetAbs        /Get absolute value if even exponent
137 913E |          JUMP End
         |                           /
138 114A |  EZero  LOAD One          /b^0 = 1  OR  1^n = 1
139 2144 |          STORE Temp
13A 913E |          JUMP End
         |                           /
13B 1141 |  EOne   LOAD Base         /b^1 = b  OR  0^n = 0
13C 2144 |          STORE Temp
13D 913E |          JUMP End
         |                           /
13E 1144 |  End    LOAD Temp
13F 6000 |          OUTPUT
140 7000 |          HALT
         |                           /
141 0000 |  Base   DEC 0
142 0000 |  Expnt  DEC 0
143 0000 |  Result DEC 0
144 0000 |  Temp   DEC 0
145 0000 |  Temp2  DEC 0
```

```
146 0000 |  Temp3  DEC 0
147 0000 |  Temp4  DEC 0
148 0000 |  Ctr1   DEC 0
149 0000 |  Ctr2   DEC 0
14A 0001 |  One    DEC 1
14B 0002 |  Two    DEC 2
         |                          /
         |                          /     Mult is a function that multiplies a positive base for
         |                          /     exponentiation.
         |                          /
14C 0000 |  Mult   HEX 0
14D 1141 |         LOAD Base
14E 414A |         SUBT One
14F 2149 |         STORE Ctr2
150 1144 |  MLoop  LOAD Temp        /Multiply by using repetitive addition
151 3145 |         ADD Temp2
152 2144 |         STORE Temp
153 1149 |         LOAD Ctr2
154 414A |         SUBT One
155 2149 |         STORE Ctr2
156 8400 |         SKIPCOND 400
157 9150 |         JUMP MLoop
158 1144 |         LOAD Temp
159 2145 |         STORE Temp2
15A C14C |         JUMPI Mult       /Return to caller
         |                          /
         |                          /     MulNeg is a function that multiplies a negative base for
         |                          /     exponentiation.
         |                          /
15B 0000 |  MulNeg HEX 0
15C 1145 |         LOAD Temp2
15D 414A |         SUBT One
15E 8800 |         SKIPCOND 800
15F 9169 |         JUMP MNEnd
160 2149 |         STORE Ctr2
161 1144 |  NLoop  LOAD Temp        /Multiply by using repetitive addition
162 3146 |         ADD Temp3
163 2144 |         STORE Temp
164 1149 |         LOAD Ctr2
165 414A |         SUBT One
166 2149 |         STORE Ctr2
167 8400 |         SKIPCOND 400
168 9161 |         JUMP NLoop
169 1144 |  MNEnd  LOAD Temp
16A 2146 |         STORE Temp3
16B C15B |         JUMPI MulNeg     /Return to caller
```

```
         |                          /
         |                          /     GetAbs calculates the absolute value of a negative value.
         |                          /
16C 0000 |  GetAbs HEX 0
16D A000 |         CLEAR
16E 2149 |         STORE Ctr2
16F 1144 |         LOAD Temp
170 314A |  GoZero ADD One          /Calculate absolute value by adding until zero
171 2144 |         STORE Temp
172 1149 |         LOAD Ctr2
173 314A |         ADD One
174 2149 |         STORE Ctr2
175 1144 |         LOAD Temp
176 8400 |         SKIPCOND 400
177 9170 |         JUMP GoZero
178 1149 |         LOAD Ctr2
179 2144 |         STORE Temp
17A 2145 |         STORE Temp2
17B C16C |         JUMPI GetAbs     /Return to caller
         |                          /
         |                          /     Div2 performs a rudimentary division of the exponent by two.
         |                          /
17C 0000 |  Div2   HEX 0
17D 1142 |         LOAD Expnt
17E 2147 |         STORE Temp4
17F 8800 |  DLoop  SKIPCOND 800     /Divide by using repetitive subtraction
180 9185 |         JUMP DEnd
181 1147 |         LOAD Temp4
182 414B |         SUBT Two
183 2147 |         STORE Temp4
184 917F |         JUMP DLoop
185 C17C |  DEnd   JUMPI Div2       /Return to caller
         |         END
```

```
Assembly successful.

          SYMBOL TABLE
          ------------------------------------------------
           Symbol | Defined | References
          --------+---------+-------------------------------
           Base   |   141   | 101, 109, 110, 11D, 127, 13B,
                  |         | 14D
           Btween |   115   | 113
           Ctr1   |   148   | 11C, 121, 123, 12B, 12D
           Ctr2   |   149   | 14F, 153, 155, 160, 164, 166,
                  |         | 16E, 172, 174, 178
           DEnd   |   185   | 180
           DLoop  |   17F   | 184
           Div2   |   17C   | 132, 185
           ELoop1 |   120   | 125
           ELoop2 |   12A   | 12F
           EOne   |   13B   | 10C, 11B
           EZero  |   138   | 10F, 117
           End    |   13E   | 126, 135, 137, 13A, 13D
           EvnExp |   136   | 134
           Expnt  |   142   | 108, 10D, 118, 17D
           GetAbs |   16C   | 106, 136, 17B
           GetExp |   107   | 105
           GoZero |   170   | 177
           Here   |   10D   | 10B
           MLoop  |   150   | 157
           MNEnd  |   169   | 15F
           MulNeg |   15B   | 12A, 16B
           Mult   |   14C   | 120, 15A
           NLoop  |   161   | 168
           NegBs  |   127   | 11F
           OddExp |   135   |
           One    |   14A   | 111, 119, 122, 12C, 138, 14E,
                  |         | 154, 15D, 165, 170, 173
           Result |   143   |
           Start  |   100   |
           Temp   |   144   | 102, 129, 131, 139, 13C, 13E,
                  |         | 150, 152, 158, 161, 163, 169,
                  |         | 16F, 171, 175, 179
           Temp2  |   145   | 103, 151, 159, 15C, 17A
           Temp3  |   146   | 128, 130, 162, 16A
           Temp4  |   147   | 17E, 181, 183
           There  |   118   | 114
           Two    |   14B   | 182
          ------------------------------------------------
```