

```

1: //V ChatGPT 1st Version (Modified for DEBUG)
2: #include <stdio.h>
3: #include <math.h>
4: #include <stdlib.h>
5:
6: //V4 improve execution time + use of string
7: int sumOfDigitsTo4(int num);
8: int digit4Pow[10];
9:
10: int main() {
11:     printf("4-digit Armstrong numbers:\n");
12:
13:     for(int i=0;i<10;i++)
14:         digit4Pow[i]=pow(i, 4);
15:
16:     for (int num = 1000; num <= 9999; ++num)
17:         if (num==sumOfDigitsTo4(num))
18:             printf("%d\n", num);
19:     return 0;
20: }
21:
22: int sumOfDigitsTo4(int num){
23:     int sum=0;
24:     char str[12];
25:     int digit;
26:     itoa(num,str,10);
27:     printf("str=%s\n",str);
28:     for(int i=0;i<4;i++){
29:         digit= str[i] - '0';
30:         sum += digit4Pow[digit];
31:         printf("sum-%d=%d\n",i,sum);
32:     }
33:     system("PAUSE");
34:
35:     // while (num > 0) {
36:     //     int digit = num % 10;
37:     //     sum += pow(digit, 4);
38:     //     sum += digit4Pow[digit];
39:     //     num /= 10;
40:     // }
41:     return sum;
42: }
43:
44:
45: ///V3 improve execution time
46: //int sumOfDigitsTo4(int num);
47: //int digit4Pow[10];
48: //
49: //int main() {
50: //    printf("4-digit Armstrong numbers:\n");
51: //
52: //    for(int i=0;i<10;i++)
53: //        digit4Pow[i]=pow(i, 4);
54: //

```

```

55: //    for (int num = 1000; num <= 9999; ++num)
56: //        if (num==sumOfDigitsTo4(num))
57: //            printf("%d\n", num);
58: //    return 0;
59: //}
60: //
61: //int sumOfDigitsTo4(num){
62: //    int sum=0;
63: //    while (num > 0) {
64: //        int digit = num % 10;
65: //        sum += pow(digit, 4);
66: //        sum += digit4Pow[digit];
67: //        num /= 10;
68: //    }
69: //    return sum;
70: //}
71:
72:
73: // V2 based on digits
74: //int sumOfDigitsTo4(int num);
75: //
76: //int main() {
77: //    int num=0;
78: //    printf("4-digit Armstrong numbers:\n");
79: //    for(int dgt4=1;dgt4<10;dgt4++)
80: //        for(int dgt3=0;dgt3<10;dgt3++)
81: //            for(int dgt2=0;dgt2<10;dgt2++)
82: //                for(int dgt1=0;dgt1<10;dgt1++){
83: //                    num=dgt4*1000+dgt3*100+dgt2*10+dgt1;
84: //                    if (num==sumOfDigitsTo4(num))
85: //                        printf("%d\n", num);
86: //                }
87: //
88: //    for (int num = 1000; num <= 9999; ++num)
89: //        if (num==sumOfDigitsTo4(num))
90: //            printf("%d\n", num);
91: //    return 0;
92: //}
93: //
94: //int sumOfDigitsTo4(num){
95: //    int sum=0;
96: //    while (num > 0) {
97: //        int digit = num % 10;
98: //        sum += pow(digit, 4);
99: //        num /= 10;
100: //    }
101: //    return sum;
102: //}
103:
104:
105: //int sumOfDigitsTo4(int num);
106: // V1.2 chatGPT code with procedural abstraction
107: //int main() {
108: //    printf("4-digit Armstrong numbers:\n");

```

```

109: //     for (int num = 1000; num <= 9999; ++num)
110: //         if (num==sumOfDigitsTo4(num))
111: //             printf("%d\n", num);
112: //     return 0;
113: //}
114: //
115: //int sumOfDigitsTo4(num){
116: //     int sum=0;
117: //     while (num > 0) {
118: //         int digit = num % 10;
119: //         sum += pow(digit, 4);
120: //         num /= 10;
121: //     }
122: //     return sum;
123: //}
124:
125: // V1.1 chatGPT code with debug code (and correction)
126: //int main() {
127: //     printf("4-digit Armstrong numbers:\n");
128: //
129: //     for (int num = 1000; num <= 9999; ++num) {
130: //         int originalNum = num;
131: //         int sum = 0;
132: //         printf("%dst iteration-Start: num:%d\torNum:%d\tsum:%d\n",
133: //             num-999,num,originalNum,sum);
134: //         while (num > 0) {
135: //             int digit = num % 10;
136: //             sum += pow(digit, 4);
137: //             num /= 10;
138: //         }
139: //         if (sum == originalNum) {
140: //             printf("%d\n", originalNum);
141: //         }
142: //         printf("%dst iteration-End: num:%d\torNum:%d\tsum:%d\n",
143: //             num-999,num,originalNum,sum);
144: //         system("pause");
145: //         num=originalNum;
146: //     }
147: //     return 0;
148: //}
149:
150: ///// V1 chatGPT code
151: //int main() {
152: //     printf("4-digit Armstrong numbers:\n");
153: //
154: //     for (int num = 1000; num <= 9999; ++num) {
155: //         int originalNum = num;
156: //         int sum = 0;
157: //         while (num > 0) {
158: //             int digit = num % 10;
159: //             sum += pow(digit, 4);
160: //             num /= 10;
161: //         }
162: //         if (sum == originalNum) {

```

```
163: //          printf("%d\n", originalNum);
164: //          }
165: //      }
166: //      return 0;
167: //}
168:
169:
```