```
1 #include <stdio.h>
 2 #include <unistd.h>
 3 #include <string.h>
 4 #include <math.h>
 5 #include <stdlib.h>
 7 #define MAX N 60
 8 #define SUCCESS 1
10 void part1(float num[], int length)
11 {
12
13
       float product;
14
       int K = 0;
15
       int flag = 0;
       int zero flag = 0;
16
       printf("The numbers are:\n");
17
       for (int i = 0; i < length ; i++)</pre>
18
19
           printf("Number %d %e \n",i + 1,num[i]);
20
21
           if (num[i] == 0)
22
                zero_flag = 1;
23
       if(zero flag == 1)
24
25
           printf("The result is zero times K to the power of 0\n");
26
27
       while(!flag)
28
29
           product = num[0];
           for(int i = 1; i < length; i++)</pre>
30
31
                product *= num[i];
32
33
            // If the numbers overflow, divide by 10 to scale
34
35
            // and retry the multiplication
           if(product == INFINITY){
36
37
                for(int i = 0; i < length; i++)</pre>
38
39
                    num[i] /= 10;
40
                K += length;
41
42
43
           else if(product == 0)
44
                for(int i = 0; i < length; i++)</pre>
45
46
                    num[i] *= 10;
47
48
49
                K -= length;
50
           else{
51
52
                flaq = 1;
53
                printf("The product is: %.6e times 10 to the power of %d \n", product,K);
54
55
           if(abs(K) == INFINITY)
56
57
                flag = 1;
58
59
       }
60
61 }
62
63 int main(void)
64 {
```

```
float num[3] = {powf(10,30), powf(10,30), powf(10,1)};
for part1(num,3);
return SUCCESS;
for part1 for powf(10,30), powf(10,30), powf(10,1);
for part1(num,3);
for powf(10,30), powf(10,30), powf(10,30), powf(10,1)};
for part1(num,3);
for part
```