```
1.  #include <stdio.h>
2.  #include <stdlib.h>
3.
4.  void
5.  add(char* to, char* from)
6.  {
7.      for(int i = 0; i < 26; i++){
8.          to[i] += from[i];
9.      }
10. }
11.
12. void
13. sub(char* to, char* from)
14. {
15.     for(int i = 0; i < 26; i++){
16.         to[i] -= from[i];
17.     }
18. }
19.
20. void
21. tosimple(int b, char* mas, char* sdef)
22. {
23.     int a = b;
24.     if(a > 1){
25.         for(int i = 2; i <= a;i++){
26.             int c = i;
27.             while(c != 1){
28.                 for(int j = 1; j < 26; j++){
29.                     if(c % sdef[j] == 0){
30.                         c /= sdef[j];
31.                         mas[j]++;
32.                         break;
33.                     }
34.                 }
35.             }
36.             mas[0] = 1;
37.         }
38.     }
39.     if(a == 0 || a == 1){
40.         mas[0] = 1;
41.     }
42. }
43.
44. int
45. bigPofk(int n, int k, int* m)
46. {
47.     char sdef[26] = {1,2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,71,73,79,83,89,97};
48.     char s1[26] = {0};
49.     char s2[26] = {0};
50.     tosimple(n,s1,sdef);
51.     for (int i = 0; i < k; ++i){
52.         for(int j = 0; j < 26; j++)
53.             s2[j] = 0;
54.         tosimple(m[i],s2,sdef);
55.         sub(s1,s2);
56.     }
57.     int res = 1;
58.     s1[0] = 1;
```

```
59.      for (int i = 0; i < 26; i++){
60.          while(s1[i]){
61.              res *= sdef[i];
62.              s1[i]--;
63.          }
64.      }
65.      return res;
66. }
67.
68. int
69. main(void)
70. {
71.      int n,k;
72.      printf("Введите n,k,mi(i=1,k)\n");
73.      scanf("%d %d",&n,&k);
74.      int m[k];
75.      for (int i = 0; i < k; i++){
76.          printf("m[%d] = ",i);
77.          scanf("%d",&m[i]);
78.      }
79.      double p[k];
80.      for (int i = 0; i < k; i++){
81.          printf("p[%d] = ",i);
82.          scanf("%lf",&p[i]);
83.      }
84.      printf("                    %d!\n",n);
85.      printf("P%d(",k);
86.      for (int i = 0; i < k-1; ++i){
87.          printf("%d,",m[i]);
88.      }
89.      printf("%d) =        ---------",m[k-1]);
90.      for (int i = 0; i < k; ++i){
91.          printf(" * %.3lf^%d",p[i],m[i]);
92.      }
93.      double res = 1;
94.      res *= bigPofk(n,k,m);
95.      for (int i = 0; i < k; i++){
96.          int z = m[i];
97.          while(z){
98.              res *= p[i];
99.              z--;
100.         }
101.     }
102.     printf(" = %lf\n",res);
103.     printf("                    ");
104.     for (int i = 0; i < k; ++i){
105.         printf("%d!",m[i]);
106.     }
107.     printf("\n");
108.     return 0;
109. }
```