

```
1. #include <stdio.h>
2. #include <stdlib.h>
3. #include <math.h>
4.
5. void
6. add(char* to, char* from)
7. {
8.     for(int i = 0; i < 26; i++){
9.         to[i] += from[i];
10.    }
11. }
12.
13. void
14. sub(char* to, char* from)
15. {
16.     for(int i = 0; i < 26; i++){
17.         to[i] -= from[i];
18.     }
19. }
20.
21. void
22. tosimple(int b, char* mas, char* sdef)
23. {
24.     int a = b;
25.     if(a > 1){
26.         for(int i = 2; i <= a; i++){
27.             int c = i;
28.             while(c != 1){
29.                 for(int j = 1; j < 26; j++){
30.                     if(c % sdef[j] == 0){
31.                         c /= sdef[j];
32.                         mas[j]++;
33.                         break;
34.                     }
35.                 }
36.             }
37.         }
38.     }
39.     mas[0] = 1;
40. }
41.
42.
43. int
44. bigC(int n, int m)
45. {
46.     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};
47.     char s1[26] = {0};
48.     char s2[26] = {0};
49.     char s3[26] = {0};
50.     tosimple(n, s1, sdef);
51.     tosimple(m, s2, sdef);
52.     tosimple(n-m, s3, sdef);
53.     add(s2, s3);
54.     sub(s1, s2);
55.     int res = 1;
56.     s1[0] = 1;
57.     for (int i = 0; i < 26; i++)
58.     {
```

```
59.         while(s1[i]){
60.             res *= sdef[i];
61.             s1[i]--;
62.         }
63.     }
64.     return res;
65. }
66.
67. double
68. bigP(int n, int m, double p)
69. {
70.     double res = 1;
71.     for(int i = 0; i < m; i++){
72.         res *= p;
73.     }
74.     for (int i = 0; i < n-m; ++i)
75.     {
76.         res *= (1-p);
77.     }
78.     res *= bigC(n,m);
79.     printf("C_from(%d)_by(%d)*%lf^%d*%lf^%d\n", n, m, p, m, 1-p, n-m);
80.     return res;
81. }
82.
83.
84. int
85. main(void)
86. {
87.     int n,m,m1,m2;
88.     double p;
89.     printf("Введите через пробел n,m,m1,m2\n");
90.     scanf("%d %d %d %d",&n,&m,&m1,&m2);
91.     if(n < m){
92.         printf("Некорретные n и m\n");
93.         return 0;
94.     }
95.     if(m1 > n || m2 > n){
96.         printf("Некорретные m1 и m2\n");
97.         return 0;
98.     }
99.     if(m2 < m1){
100.         printf("m1 и m2 поменялись местами\n");
101.         int tmp = m1;
102.         m1 = m2;
103.         m2 = tmp;
104.     }
105.     printf("Введите p\n");
106.     scanf("%lf",&p);
107.     if(p > 1 || p < 0){
108.         printf("Некорретная вероятность p\n");
109.         return 0;
110.     }
111.     int num;
112.     printf("Выберите вариант:\n");
113.     printf("1 - P%d(k = %d)\n", n, m);
114.     printf("2 - P%d(k < %d)\n", n, m);
115.     printf("3 - P%d(k >= %d)\n", n, m);
116.     printf("4 - P%d( %d <= k <= %d)\n", n, m1, m2);
```

```
117. printf("Любой другой - все варианты\n");
118. scanf("%d",&num);
119. double res;
120. int flg = 1;
121. switch(num){
122.     default:
123.         flg = 0;
124.         case 1:
125.             printf("_____ \n");
126.             printf("P%d(k = %d) =\n",n,m);
127.             res = bigP(n,m,p);
128.             printf(" =%- .5lf\n",res);
129.             if(flg)
130.                 break;
131.         case 2:
132.             printf("_____ \n");
133.             printf("P%d(k < %d) =\n",n,m);
134.             res = 0;
135.             for(int i = 0; i < m; i++){
136.                 res += bigP(n,i,p);
137.                 if(i != m-1)
138.                     printf(" + \n");
139.             }
140.             printf(" = %- .5lf\n",res);
141.             if(flg)
142.                 break;
143.         case 3:
144.             printf("_____ \n");
145.             printf("P%d(k >= %d)\n",n,m);
146.             res = 0;
147.             for (int i = m; i <= n ; i++){
148.                 res += bigP(n,i,p);
149.                 if(i != n)
150.                     printf(" + \n");
151.             }
152.             printf(" = %- .5lf\n",res);
153.             if(flg)
154.                 break;
155.         case 4:
156.             printf("_____ \n");
157.             printf("P%d( %d <= k <= %d)\n",n,m1,m2);
158.             res = 0;
159.             for (int i = m1; i <= m2 ; i++){
160.                 res += bigP(n,i,p);
161.                 if(i != m2)
162.                     printf(" + \n");
163.             }
164.             printf(" = %- .5lf\n",res);
165.             break;
166.     }
167.     return 0;
168. }
```