

```
1. #include <stdlib.h>
2. #include <stdio.h>
3. #include <fcntl.h>
4. #include <math.h>
5. #include <string.h>
6.
7. struct dpoint{
8.     double a;
9.     double b;
10.    double cent;
11.    int n;
12.    double f;
13. };
14.
15. int
16. generateMas(struct dpoint* mas, char* name)
17. {
18.     int n=0;
19.     FILE* in = fopen(name, "r");
20.     n = 0;
21.     while(fscanf(in, "%lf %lf %d", &mas[n].a, &mas[n].b, &mas[n].n) != EOF){
22.         n++;
23.     }
24.     fclose(in);
25.     return n;
26. }
27.
28. void
29. HandleInput(struct dpoint* mas, int n)
30. {
31.     for (int i = 0; i < n; i++){
32.         printf("(a,b),n [%d]:\n", i);
33.         scanf("%lf %lf %d", &mas[i].a, &mas[i].b, &mas[i].n);
34.     }
35. }
36.
37. void
38. genGistDataAbs(struct dpoint* mas, int n)
39. {
40.     FILE* out = fopen("Abs.txt", "w");
41.     for (int i = 0; i < n; i++){
42.         fprintf(out, "0 %.3lf\n", mas[i].a);
43.         fprintf(out, "%d %.3lf\n", mas[i].n, mas[i].a);
44.         fprintf(out, "%d %.3lf\n", mas[i].n, mas[i].b);
45.         fprintf(out, "0 %.3lf\n", mas[i].b);
46.     }
47.     fclose(out);
48. }
49.
50. void
51. genGistDataRel(struct dpoint* mas, int n)
52. {
53.     FILE* out = fopen("Rel.txt", "w");
54.     for (int i = 0; i < n; i++){
55.         fprintf(out, "0 %.3lf\n", mas[i].a);
56.         fprintf(out, "%.3lf %.3lf\n", mas[i].f, mas[i].a);
57.         fprintf(out, "%.3lf %.3lf\n", mas[i].f, mas[i].b);
58.         fprintf(out, "0 %.3lf\n", mas[i].b);
```

```
59.     }
60.     fclose(out);
61. }
62.
63. void
64. genPolDataAbs(struct dpoint* mas, int n)
65. {
66.     FILE* out = fopen("Abs.txt", "w");
67.     for (int i = 0; i < n; ++i){
68.         fprintf(out, "%d %.3lf\n", mas[i].n, mas[i].cent);
69.     }
70.     fclose(out);
71. }
72.
73. void
74. genPolDataRel(struct dpoint* mas, int n)
75. {
76.     FILE* out = fopen("Rel.txt", "w");
77.     for (int i = 0; i < n; ++i){
78.         fprintf(out, "%.3lf %.3lf\n", mas[i].f, mas[i].cent);
79.     }
80.     fclose(out);
81. }
82.
83. void
84. genFDataInt(struct dpoint* mas, int n)
85. {
86.     FILE* out = fopen("FInt.txt", "w");
87.     double sum = 0.0;
88.     fprintf(out, "0 %lf\n", -2*mas[0].a);
89.     fprintf(out, "0 %lf\n\n", mas[0].a);
90.     for (int i = 0; i < n; ++i){
91.         sum += mas[i].f;
92.         fprintf(out, "%.3lf %.3lf\n", sum, mas[i].a);
93.         fprintf(out, "%.3lf %.3lf\n\n", sum, mas[i].b);
94.     }
95.     fprintf(out, "1.000 %.3lf\n", mas[n-1].b);
96.     fprintf(out, "1.000 %.3lf\n", 2*mas[n-1].b);
97.     fclose(out);
98. }
99.
100. void
101. genFDataGroup(struct dpoint* mas, int n)
102. {
103.     FILE* out = fopen("FGroup.txt", "w");
104.     double sum = 0.0;
105.     fprintf(out, "0 %lf\n", -2*mas[0].cent);
106.     fprintf(out, "0 %lf\n\n", mas[0].cent);
107.     for (int i = 0; i < n-1; ++i){
108.         sum += mas[i].f;
109.         fprintf(out, "%.3lf %.3lf\n", sum, mas[i].cent);
110.         fprintf(out, "%.3lf %.3lf\n\n", sum, mas[i+1].cent);
111.     }
112.     fprintf(out, "1.000 %.3lf\n", mas[n-1].cent);
113.     fprintf(out, "1.000 %.3lf\n", 2*mas[n-1].cent);
114.     fclose(out);
115. }
116.
```

```
117. void
118. genDigParam(struct dpoint* mas, int n)
119. {
120.     FILE* out = fopen("Dig.txt", "w");
121.     double x = 0.0;
122.     double D = 0.0;
123.     double sig = 0.0;
124.     double S = 0.0;
125.     int sum = 0;
126.     for (int i = 0; i < n; ++i){
127.         x+= mas[i].cent*mas[i].n;
128.         D+= mas[i].cent*mas[i].n*mas[i].cent;
129.         sum += mas[i].n;
130.     }
131.     x /= sum;
132.     D /= sum;
133.     D -= x*x;
134.     sig = sqrt(D);
135.     S = D*(sum-1)/sum;
136.     fprintf(out, "X выборочное = %- .6lf\n", x);
137.     fprintf(out, "D выб = %- .6lf\n", D);
138.     fprintf(out, "среднее кв.отклонение = %- .6lf\n", sig);
139.     fprintf(out, "S = %- .6lf\n", S);
140.     fclose(out);
141. }
142.
143. void
144. delData(int tmp)
145. {
146.     switch(tmp){
147.         case 1:
148.             system("rm Abs.png");
149.             system("rm Rel.png");
150.             system("rm Abs.txt");
151.             system("rm Rel.txt");
152.             break;
153.         case 2:
154.             system("rm Abs.png");
155.             system("rm Rel.png");
156.             system("rm Abs.txt");
157.             system("rm Rel.txt");
158.             break;
159.         case 3:
160.             system("rm FGroup.png");
161.             system("rm FGroup.txt");
162.             system("rm FInt.png");
163.             system("rm FInt.txt");
164.             break;
165.         case 4:
166.             system("rm Dig.txt");
167.             break;
168.         default:
169.             break;
170.     }
171. }
172.
173. int
174. main(void)
```

```
175. {
176.     struct dpoint mas[100];
177.     int n = 0;
178.     int tmp;
179.     printf("0 - ввод данных их файла\n");
180.     printf("1 - ручной ввод\n");
181.     scanf("%d",&tmp);
182.     if(tmp){
183.         printf("Число интервалов n\n");
184.         scanf("%d",&n);
185.         HandleInput(mas,n);
186.     }else{
187.         n = generateMas(mas,"data.txt");
188.     }
189.     int sum = 0;
190.     for (int i = 0; i < n; i++){
191.         sum += mas[i].n;
192.     }
193.     for (int i = 0; i < n; i++){
194.         mas[i].f = (double)mas[i].n/sum;
195.         mas[i].cent = (mas[i].a + mas[i].b)/2;
196.     }
197.     printf("1 - интервальный ряд частот и относительных частот(гистограммы)\n");
198.     printf("2 - группированный ряд частот и относительных частот(полигоны)\n");
199.     printf("3 - постоение F(x) \n");
200.     printf("4 - вычисление характеристик(Xв,Dв,sig,S\n");
201.     scanf("%d",&tmp);
202.     switch(tmp){
203.         case 1:
204.             genGistDataAbs(mas,n);
205.             genGistDataRel(mas,n);
206.             system("gnuplot scr_1_1.txt");
207.             system("gnuplot scr_1_2.txt");
208.             system("ristretto Abs.png");
209.             break;
210.         case 2:
211.             genPolDataAbs(mas,n);
212.             genPolDataRel(mas,n);
213.             system("gnuplot scr_1_1.txt");
214.             system("gnuplot scr_1_2.txt");
215.             system("ristretto Abs.png");
216.             break;
217.         case 3:
218.             genFDataInt(mas,n);
219.             genFDataGroup(mas,n);
220.             system("gnuplot scr_3_1.txt");
221.             system("gnuplot scr_3_2.txt");
222.             system("ristretto FInt.png");
223.             break;
224.         case 4:
225.             genDigParam(mas,n);
226.             system("subl Dig.txt");
227.             break;
228.         default:
229.             break;
230.     }
231.     printf("Введите целое число для завершения\n");
232.     scanf("%d",&n);
```

```
233.     delData(tmp);  
234.     return 0;  
235. }
```