

## 1 0-1 之间的随机数均匀性测试

本实验假设所有代码文件放在同一文件夹中。请按照要求，产生可执行文件，并给出每一个步骤的截图。以下要求使用 Linux 或者 Windows 的 GCC 工具。

### 1.1 均匀性测试

以下包括代码和操作。

#### 1.1.1 main1.c 文件

---

```
1 #include <stdio.h>
2 #include <time.h> // 支持time()函数
3 #include <stdlib.h> // 支持伪随机数操作
4
5 int main()
6 {
7     srand(time(NULL));
8     printf("RAND_MAX: %d\n", RAND_MAX);
9     int divisor = 1024; // 用于获得一定整数区间内的随机数
10    long long sequence_length = 10; // 随机序列的长度
11    long long counters[11] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
12    // 0号单元不用
13    for(int j = 0; j < sequence_length; j++) //
14        循环sequence_length次，从而产生多个随机数
15    {
16        double d; // 用于保存[0,1)中的随机数
17        int n = rand() % divisor; // 用于获取[0,
18        divisor)之间的随机整数
19        d = n / (double)divisor; // 归一化
20        if (d < 0.1)
21        {
22            counters[1]++;
23        }
24        else if (d < 0.2)
25        {
26            counters[2]++;
27        }
28        // ... (other counters) ...
29    }
```

```
24     }
25     else if (d < 0.3)
26     {
27         counters[3]++;
28     }
29     else if (d < 0.4)
30     {
31         counters[4]++;
32     }
33     else if (d < 0.5)
34     {
35         counters[5]++;
36     }
37     else if (d < 0.6)
38     {
39         counters[6]++;
40     }
41     else if (d < 0.7)
42     {
43         counters[7]++;
44     }
45     else if (d < 0.8)
46     {
47         counters[8]++;
48     }
49     else if (d < 0.9)
50     {
51         counters[9]++;
52     }
53     else // .9 < d < 1
54     {
55         counters[10]++;
56     }
57 }
58
59 // 输出区间
60 printf("for intervals ");
61 for(int j = 1; j <= 9; j++)
62 {
```

```
63     printf("[0.%.2d, 0.%.2d)\t", j - 1, j);
64 }
65 printf("[0.9, 1.0)\n");
66
67 // 输出落入各区间中的整数的个数
68 printf("counters:\n");
69 for(int j = 1; j <= 10; j++)
70 {
71     printf("%lld\t", counters[j]);
72 }
73 printf("\n");
74
75 // 输出落入各区间中的整数的频率
76 printf("frequencies:\n");
77 for(int j = 1; j <= 10; j++)
78 {
79     printf("%lf\t", counters[j] / (double)sequence_length);
80 }
81 printf("\n");
82 return 0;
83 }
```

---

1. 根据以上代码产生可执行文件，并把产生的文件命名为 `main1.exe`。运行 `main1.exe`，指出落入不同区间的浮点数的频数和频率。
2. 把第 10 行的 10 先后改为 100, 1000, 10000，重新统计上述频数和频率。结合上一步和这一步中，你得到什么结论？
3. 把第 9 行的 1024 先后改为 2048, 4096, 8192，重新统计上述频数和频率。结合第一步和这一步中，你得到什么结论？

## 1.2 实验报告写作要求

1. 步骤详细；
2. 表述简明；
3. 图文并茂；
4. 逻辑流畅。