

第一章 顺序结构

第14课 四舍六入五留双

《信息学奥赛一本通·编程启蒙 C++版》

用 printf 函数输出保留小数位数的实型数据时，用的是四舍六入五留双原则而不是四舍五入原则，因为四舍六入五成双是一种比较精确比较科学的计数保留法，是一种数字修约规则。

对于位数很多的近似数，当有效位数确定后，其后面多余的数字应该舍去，只保留有效数字最末一位，这种修约(舍入)规则是“四舍六入五成双”，也即“4舍6入5凑偶”这里“四”是指 ≤ 4 时舍去，“六”是指 ≥ 6 时进上，“五”指的是根据5后面的数字来定，当5后有数时，舍5入1；当5后无数或为0时，需要分两种情况来讲：①5前为奇数，舍5入1；②5前为偶数，舍5不进。具体规则如下：

(1) 被修约的数字等于或小于4时，该数字舍去；

(2) 被修约的数字等于或大于6时，则进位；

(3) 被修约的数字等于5时，要看5前面的数字，若是奇数则进位，若是偶数则将5舍掉，即修约后末尾数字都成为偶数；若5的后面还有不为“0”的任何数，则此时无论5的前面是奇数还是偶数，均应进位。

举例：

1.625，用上述规则保留 2 位小数，结果是 1.62

1.6251，用上述规则保留 2 位小数，结果是 1.63

0.9375，用上述规则保留 3 位小数，结果是 0.938

0.93751，用上述规则保留 3 位小数，结果是 0.938

从统计学的角度，“四舍六入五成双”比“四舍五入”要科学，在大量运算时，它使舍入后的结果误差的均值趋于零，而不是像四舍五入那样逢五就入，导致结果偏向大数，使得误差产生积累进而产生系统误差，“四舍六入五成双”使测量结果受到舍入误差的影响降到最低。

例如： $0.15+0.25+0.35+0.45=1.2$ ，

若按四舍五入取一位小数计算： $0.2+0.3+0.4+0.5=1.4$

按“四舍六入五成双”计算， $0.2+0.2+0.4+0.4=1.2$ ，

舍入后的结果更能反映实际结果。

【例 14.1】输出浮点数

【题目描述】

读入一个双精度浮点数，分别按输出格式“%f”，“%f”保留 5 位小数，“%e”和“%g”的形式输出这个整数，每次在单独一行上输出。

【输入格式】

一个双精度浮点数。

【输出格式】

第一行是按“%f”输出的双精度浮点数；

第二行是按“%f”保留 5 位小数输出的双精度浮点数；

第三行是按“%e”输出的双精度浮点数；

第四行是按“%g”输出的双精度浮点数。

【样例输入】

12.3456789

【样例输出】

12.345679

12.34568

1.234568e+001

12.3457

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. double d;
4. int main(){
5.     cin>>d;
6.     printf("%f\n",d);
7.     printf("%.5f\n",d);
8.     printf("%e\n",d);
9.     printf("%g\n",d);
10.    return 0;
11.}
```

%f, %e 和 %g 三个区别是:

(1) %f 是以小数的形式输出，整数部分原样输出，小数点后输出 6 位小数。如要保留其他位数的小数，就需要设置，如第 7 行代码中表示保留 5 位小数。但在实际编写的过程中，使用 %lf 比较多。

(2) %e 是以指数形式的浮点数的格式输出

(3) %g 是自动选择宽度小的那种格式输出，而且不输出无意义的零。

【例 14.2】 四舍六入五留双

【题目描述】

输入一个实数 f ， 和一个位数 d

输出实数 f ， 在保留 d 位小数下的结果， 采用四舍六入五留双的近似。

f 至多有 30 位小数。

$$0 \leq f \leq 1$$

$$1 \leq d \leq 8$$

所谓四舍六入五留双，是指如果恰好是 0.5 的情况，会把他近似到使得前一位是偶数。

比如近似到整数，0.4 为 0，0.5 为 0，0.50001 为 1，0.6 为 1，1.5 为 2，2.5 为 2。

【输入格式】

一行一个浮点数 f 和一个位数 d 数。

【输出格式】

一行一个浮点数表示答案。

【样例输入】

0.123456789 5

【样例输出】

0.12346

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int main(){
4.     double x;
5.     int y;
6.     scanf("%lf",&x);
7.     scanf("%d",&y);
8.     printf("%.*lf\n",y,x);
9.     return 0;
10. }
```

【例 14.3】 数学课上

【题目描述】

给出一个浮点数，怎么判断这个数离前后相邻两个整数哪个更近，输出距离更近的整数。请你按照四舍五入原则 编程输出这个数。

【输入格式】

输入一行，包含 1 个数： n ($0.0 \leq n \leq 100000.0$) ,表示题目要求输入的浮点数。题目保证输入浮点数小数点后保留最多 8 位。

【输出格式】

输出共计 1 行，包含 1 个数，表示题目所求的距离更近的整数。

【样例输入】

4.5

【样例输出】

5

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. double n;
4. int m;
5. int main(){
6.     scanf("%lf",&n);
7.     n+=0.5;
8.     m=n;
9.     printf("%d",m);
10.    return 0;
11.}
```

练 14.1 歌手大奖赛

【题目描述】

歌手大奖赛上 6 名评委给一位参赛者打分，6 个人打分的平均分为 9.6 分；如果去掉一个最高分，这名参赛者的平均分为 9.4 分；如果去掉一个最低分，这名参赛者的平均分为 9.8 分；如果去掉一个最高分和一个最低分，这名参赛者的平均是多少？

【输入格式】

无

【输出格式】

使用`%5.2f` 按实数格式输出，保留 2 位小数。

【样例输入】

无

【样例输出】

无

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. double maxx,minx;
4. int main(){
5.     maxx=6*9.6-5*9.4;
6.     minx=6*9.6-5*9.8;
7.     printf("%5.2lf", (6*9.6-maxx-minx)/4);
8.     return 0;
9. }
```

练 14.2 平均分

【题目描述】

已知某班有男同学 x 位，女同学 y 位， x 位男生平均分是 87 分， y 位女生的平均分是 85，问全体同学平均分是多少分。

【输入格式】

男女同学人数。

【输出格式】

平均分（保留 4 位小数）。

【样例输入】

2 3

【样例输出】

85.8000

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int x,y;
4. int main(){
5.     scanf("%d%d",&x,&y);
6.     printf("%.4lf",1.0*(x*87+y*85)/(x+y));
7.     return 0;
8. }
```

练 14.3 地球人口承载力估计

【题目描述】

假设地球上的新生资源按恒定速度增长。照此测算，地球上现有资源加上新生资源可供 x 亿人生活 a 年，或供 y 亿人生活 b 年。

为了能够实现可持续发展，避免资源枯竭，地球最多能够养活多少亿人？

【输入格式】

一行，包括四个正整数 x ， a ， y ， b ，两个整数之间用单个空格隔开。 $x > y$ ， $a < b$ ， $ax < by$ ，各整数均不大于 10000

【输出格式】

一个实数 zz ，表示地球最多养活 zz 亿人，舍入到小数点后两位

【样例输入】

110 90 90 210

【样例输出】

75.00

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int a,x,b,y;
4. double ans;
5. int main(){
6.     cin>>x>>a>>y>>b;
7.     ans=1.0*(x*a-y*b)/(a-b);
8.     printf("%.2lf",ans);
9.     return 0;
10.}
```

练 14.4 计算多项式的值

【题目描述】

对于多项式 $f(x)=ax^3+bx^2+cx+d$ 和给定的 a,b,c,d,x , 计算 $f(x)$ 的值, 保留到小数点后 7 位。

【输入格式】

输入仅一行, 包含 5 个实数, 分别是 x , 及参数 a 、 b 、 c 、 d 的值, 每个数都是绝对值不超过 100 的双精度浮点数。数与数之间以一个空格分开。

【输出格式】

输出一个实数, 即 $f(x)$ 的值, 保留到小数点后 7 位。

【样例输入】

2.31 1.2 2 2 3

【样例输出】

33.0838692

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. double x,a,b,c,d,ans;
4. int main()
5. {
6.     cin>>x>>a>>b>>c>>d;
7.     ans=a*x*x*x+b*x*x+c*x+d;
8.     cout<<fixed<<setprecision(7)<<ans;
9.     return 0;
10.}
```

谢谢！

