代码部分：

1：读入：先读入一个正整数n，然后连续读入n个整数

2：存储：分别存储到vector和set中

3：输出：输出读入的原数组，按升序排序的不去重数组，去重数组

4：测试：从文件读入测试数据（不合法输入部分用控制台测试），与正解进行比较

测试部分：

1：不合法读入：

|  |  |  |  |
| --- | --- | --- | --- |
| 数据 | 类型 | 正确解 | 结果 |
| N=-1 | N<=0 | 报错 | 报错 |
| N=heubchrhhf | N不为整数 | 报错 | 报错 |
| 某个整数为3.14 | 不为整数 | 报错 | 报错 |

2：无重复数据

2.1：原先有序的数据

|  |  |  |  |
| --- | --- | --- | --- |
| 数据 | 类型 | 正解 | 结果 |
| N=3  1 2 3 | 全是自然数 | 1 2 3  1 2 3 | 1 2 3  1 2 3 |
| N=5  -5 -4 -3 -2 -1 | 全是负数 | -5 -4 -3 -2 -1  -5 -4 -3 -2 -1 | -5 -4 -3 -2 -1  -5 -4 -3 -2 -1 |
| N=6  -6 -4 -2 0 1 102 | 正负都有 | -6 -4 -2 0 1 102  -6 -4 -2 0 1 102 | -6 -4 -2 0 1 102  -6 -4 -2 0 1 102 |

2.2：原先反序的数据

|  |  |  |  |
| --- | --- | --- | --- |
| 数据 | 类型 | 正解 | 结果 |
| N=3  3 2 1 | 全是自然数 | 1 2 3  1 2 3 | 1 2 3  1 2 3 |
| N=5  -1 -2 -3 -4 -5 | 全是负数 | -5 -4 -3 -2 -1  -5 -4 -3 -2 -1 | -5 -4 -3 -2 -1  -5 -4 -3 -2 -1 |
| N=6  102 1 0 -2 -4 -6 | 正负都有 | -6 -4 -2 0 1 102  -6 -4 -2 0 1 102 | -6 -4 -2 0 1 102  -6 -4 -2 0 1 102 |

2.3：原先乱序的数据

|  |  |  |  |
| --- | --- | --- | --- |
| 数据 | 类型 | 正解 | 结果 |
| N=3  2 3 1 | 全是自然数 | 1 2 3  1 2 3 | 1 2 3  1 2 3 |
| N=5  -3 -5 -1 -4 -2 | 全是负数 | -5 -4 -3 -2 -1  -5 -4 -3 -2 -1 | -5 -4 -3 -2 -1  -5 -4 -3 -2 -1 |
| N=6  0 102 1 -4 -2 -6 | 正负都有 | -6 -4 -2 0 1 102  -6 -4 -2 0 1 102 | -6 -4 -2 0 1 102  -6 -4 -2 0 1 102 |

3：有重复数据

3.1：原先有序的数据

|  |  |  |  |
| --- | --- | --- | --- |
| 数据 | 类型 | 正解 | 结果 |
| N=7  1 2 3 3 4 4 7 | 全是自然数 | 1 2 3 3 4 4 7  1 2 3 4 7 | 1 2 3 3 4 4 7  1 2 3 4 7 |
| N=9  -5 -4 -4 -4 -3 -2 -2 -1 -1 | 全是负数 | -5 -4 -4 -4 -3 -2 -2 -1 -1  -5 -4 -3 -2 -1 | -5 -4 -4 -4 -3 -2 -2 -1 -1  -5 -4 -3 -2 -1 |
| N=8  -6 -4 -2 0 0 1 1 102 | 正负都有 | -6 -4 -2 0 0 1 1 102  -6 -4 -2 0 1 102 | -6 -4 -2 0 0 1 1 102  -6 -4 -2 0 1 102 |

3.2：原先反序的数据

|  |  |  |  |
| --- | --- | --- | --- |
| 数据 | 类型 | 正解 | 结果 |
| N=5  3 2 1 1 1 | 全是自然数 | 1 1 1 2 3  1 2 3 | 1 1 1 2 3  1 2 3 |
| N=9  -1 -2 -2 -2 -3 -4 -5 -5 -5 | 全是负数 | -5 -5 -5 -4 -3 -2 -2 -2 -1  -5 -4 -3 -2 -1 | -5 -5 -5 -4 -3 -2 -2 -2 -1  -5 -4 -3 -2 -1 |
| N=8  102 102 1 1 0 -2 -4 -6 | 正负都有 | -6 -4 -2 0 1 1 102 102  -6 -4 -2 0 1 102 | -6 -4 -2 0 1 1 102 102  -6 -4 -2 0 1 102 |

3.3：原先乱序的数据

|  |  |  |
| --- | --- | --- |
| 数据 | 正解 | 结果 |
| N=7  3 21 -3 482 482 3 3 | -3 3 3 3 21 482 482  -3 3 21 482 | -3 3 3 3 21 482 482  -3 3 21 482 |
| N=4  0 0 -7 8 | -7 0 0 8  -7 0 8 | -7 0 0 8  -7 0 8 |
| N=9  33 21 -8 0 -33 12 33 0 -8 | -33 -8 -8 0 0 12 21 33 33  -33 -8 0 12 21 33 | -33 -8 -8 0 0 12 21 33 33  -33 -8 0 12 21 33 |
| N=7  384 13 41 44 12 4 12 | 4 12 12 13 41 44 384  4 12 13 41 44 384 | 4 12 12 13 41 44 384  4 12 13 41 44 384 |
| N=5  -2 -2 -293 -293 -4 | -293 -293 -4 -2 -2  -293 -4 -2 | -293 -293 -4 -2 -2  -293 -4 -2 |

4：全都是重复数据

|  |  |  |  |
| --- | --- | --- | --- |
| 数据 | 类型 | 正解 | 结果 |
| N=3  1 1 1 | 全是正数 | 1 1 1  1 | 1 1 1  1 |
| N=5  -5 -5 -5 -5 -5 | 全是负数 | -5 -5 -5 -5 -5  -5 | -5 -5 -5 -5 -5  -5 |
| N=6  0 0 0 0 0 0 | 0 | 0 0 0 0 0 0  0 | 0 0 0 0 0 0  0 |