1. 选择题

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **C** | **BD** | **C** | **A** | **C** | **B** | **D** | **D** | **B** | **AD** |
| **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** | **19** | **20** |
| C | A | C | B | C | A | D | C | A | A |

1. 读程序题
   1. 6分

3

7

13

26

25

* 1. 8分

Improper score value

finally g()

-1

finally g()

finally g()

10

finally g()

Caught OtherException

Successful execution

* 1. 8分

int in S: 20

double in T: 20.0

double in T: 3.5

double in T: 3.5

* 1. 8分

第5天有1个桃子

第4天有3个桃子

第3天有7个桃子

第2天有15个桃子

第一天，摘的桃子数：31

31

1. 写程序题
   1. 5分

public static void main(String[] args)

{

int num = 0;

for(int i = 10; i <= 99; i++)

{

if(i % 5 == 0 && i % 3 != 0)

{

System.out.print(i);

num++;

if(num == 6)

{

System.out.println();

num = 0;

}

}

}

}

评分细则：（1）如果没有换行，而基本流程正确，扣1分；（2）如果输出的条件逻辑不对，而基本流程正确，扣2分；

* 1. 12分

public String get() 2分

{

return description;

}

public void set(String description)

{

this->description = description;

}

public double volume() 6分

{

return length \* width \* height;

}

public double volume()

{

return 4 \* Math.PI \* r \* r \* r /3;

}

for (int index = 1; index < s.length; index++) 4分

{

3DShape key = s[index].volume();

int position = index;

// Shift larger values to the right

while(position > 0 && key.volume()>s[position-1]) .volume())

{

s[position] = s[position-1];

position--;

}

s[position] = key;

}

* 1. 13分

4分

ListNode node = new ListNode ();

node.data = newData;

node.link = null;

ListNode current;

if (head == null)

head = node;

else

{

current = head;

while (current.link != null && newdata >current.link.data)

node.link = current.link;

current.link = node;

}

4分

ListNode current = head;

if(head == null)

return 0;

else

if(head.data == target)

{

head = null;

return 1;

}

else

{

while(current.link != null && current.link.data != target)

current = current.link;

if(current.link != null)

{

current.link = current.link.link;

return 1;

}else

return 0;

}

5分

if(head == null)

return;

if(head.link == null)

return;

ListNode current = head.link;

while(current != null)

{

ListNode temp = current.link;

current.link = head;

head = current;

current = temp;

}