## JAVASE(03-2.常用运算符)

试卷得分：90分

多选题

1.请选择控制台输出结果（ ）。  
  double b = 10;  
  b += b++;   
  System.out.println(++b);

 20  
 21  
 20.0  
 21.0

作答结果：✓

正解：D

解析：

2.请选择控制台输出结果（ ）。  
  float f = 20;  
  f -= --f;   
  System.out.println(f--);

 10  
 9.0  
 1.0  
 0

作答结果：✓

正解：C

解析：

3.请选择控制台输出结果（ ）。  
  int a = 10;  
  a = (a ＞ 10)? (a \*= 2):( a/= 2);  
  System.out.println(a++);

 5  
 6  
 20  
 21

作答结果：✓

正解：A

解析：

4.请选择控制台输出结果（ ）  
  int a = 6;  
  a = (a != 6.0)? (a%2):( a%4);  
  System.out.println(--a);

 -1  
 1  
 0  
 2

作答结果：✓

正解：B

解析：

5.请选择控制台输出结果（ ）。   
  int a  = 3;  
  a =((a++ ＞3) & (--a＜3))? a++ : a--;  
  System.out.println(a);

 1  
 2  
 3  
 4

作答结果：☓

正解：C

解析：

6.请选择控制台输出结果（ ）。   
  int b  = 5;  
  b =((b-- ＞3) && (--b＜3))? b++ : b--;  
  System.out.println(b);

 1  
 2  
 3  
 4

作答结果：✓

正解：C

解析：

7.请选择控制台输出结果（ ）。  
   int a = 1;  
   System.out.println(( a++＞2 || a--＜2 ));

 1  
 2  
 true  
 false

作答结果：✓

正解：D

解析：

8.请选择控制台输出结果（ ）  
  int b = 5;  
  System.out.println(( ++b＞6 || --b＜6 ));

 5  
 6  
 true  
 false

作答结果：✓

正解：C

解析：

9.请选择控制台输出结果（ ）  
  String str = "123";  
  int a =  1;  
  System.out.println(str +1 );

 124  
 1123  
 1231  
 1

作答结果：✓

正解：C

解析：

10.10. 请选择控制台输出结果（ ）。 D  
   String str = "1";  
   System.out.println(str +null );  
   

 0  
 1  
 null  
 1null

作答结果：✓

正解：D

解析：

## JAVASE(04-1.流程控制及循环)

试卷得分：93分

多选题

1.请选择控制台输出结果（ ）  
  int i =1;  
  if(i＞2){  
   i++;  
  }else if(i＞1){  
   i += 2;  
  }else{  
   i += 3;  
  }  
  System.out.println(i);

 1  
 2  
 3  
 4

作答结果：✓

正解：D

解析：

2. 请选择控制台输出结果（ ）  
  for(int i = 0;i ＜ 10;i++){  
   if(i%2 == 0){  
   i += 3;  
   }else{  
   i += 4;  
   }  
   System.out.println(i);  
  }

 打印两次分别为4,8  
 打印三次分别为4,8,12  
 打印三次分别为3,7,11  
 打印两次分别为5,9

作答结果：✓

正解：C

解析：

3.请选择控制台输出结果（ ）  
  int sum = 1;  
  for(int i = 0;i ＜ 10;i++){  
   sum += i;   
  }  
  System.out.println(sum);

 45  
 46  
 1  
 10

作答结果：✓

正解：B

解析：

4.请选择控制台输出结果（ ）  
  int max = 6;  
  for(int i = 0;i ＜ 5;i++){  
   if(max ＜ i){  
   max = i ;  
   }  
  }  
  System.out.println(max);

 1  
 2  
 5  
 6

作答结果：✓

正解：D

解析：

5.请选择控制台输出结果（ ）  
  int total = 0;  
  for(int i=1;i＜=10;i++){  
   if(i%2==0){  
   total +=i;  
   }  
  }  
  System.out.println(total);

 10  
 45  
 55  
 30

作答结果：✓

正解：D

解析：

6.请选择控制台输出结果（ ）  
  int total = 0;  
  for(int i=1;i＜=10;i++){  
   if(i%2==1){  
   total +=i;  
   }  
  }  
  System.out.println(total);

 10  
 25  
 45  
 30

作答结果：✓

正解：B

解析：

7.请选择控制台输出结果（ ）  
  int total = 0;  
  for(int i=1;i＜=10;i++){  
   if(i%2==1){  
   total -= i;  
   }else{  
              total += i;  
         }  
  }  
  System.out.println(total);

 5  
 10  
 15  
 20

作答结果：✓

正解：A

解析：

8.请选择控制台输出结果（ ）  
  int sum  = 0;  
  for(int i=0;i＜10;i++){  
   sum += i;  
   for(;i＜5;i++){  
   sum -= i;  
   }  
  }  
  System.out.println(sum);

 5  
 10  
 20  
 15

作答结果：✓

正解：C

解析：

9.请选择①处代码,使得运行控制输出99乘法表（ ）  
  for(int i =0;i＜10;i++){  
   for(int j = 0;j＜=i;j++){  
   System.out.print(\_\_\_\_①\_\_\_\_);  
   }  
   System.out.println();  
  }  
    
  控制台输出如下图案  
  0\*0=0   
  1\*0=0 1\*1=1   
  2\*0=0 2\*1=2 2\*2=4   
  3\*0=0 3\*1=3 3\*2=6 3\*3=9   
  4\*0=0 4\*1=4 4\*2=8 4\*3=12 4\*4=16   
  5\*0=0 5\*1=5 5\*2=10 5\*3=15 5\*4=20 5\*5=25   
  6\*0=0 6\*1=6 6\*2=12 6\*3=18 6\*4=24 6\*5=30 6\*6=36   
  7\*0=0 7\*1=7 7\*2=14 7\*3=21 7\*4=28 7\*5=35 7\*6=42 7\*7=49   
  8\*0=0 8\*1=8 8\*2=16 8\*3=24 8\*4=32 8\*5=40 8\*6=48 8\*7=56 8\*8=64   
  9\*0=0 9\*1=9 9\*2=18 9\*3=27 9\*4=36 9\*5=45 9\*6=54 9\*7=63 9\*8=72 9\*9=81

 i+"\*"+j+"="+i\*j+" "  
 i+"\*"+j+"="+i\*j  
 "i"+\*+"j"+"="+i\*j+" "  
 i+"\*"+j+"="+"i\*j"

作答结果：✓

正解：A

解析：

10.请选择①处代码,使得运行控制输出左直角三角形（ ）  
   for(int i =0;i＜=5;i++){  
    for(\_\_\_①\_\_\_){  
    System.out.print("\*");  
    }  
    System.out.println();  
   }  
   控制台输出如下图案  
   \*  
   \*\*  
   \*\*\*  
   \*\*\*\*  
   \*\*\*\*\*

 int j = 0;j＜=i;j++  
 int j = 0;j＜i;j++  
 int j = 1;j＜i;j++  
 int j = 0;j＜=i;j--

作答结果：✓

正解：B

解析：

11.   public class Run {  
       public static void \_\_main\_\_\_(String[] args) {  
    //题目：有1、2、3、4个数字，能组成多少个互不相同的三位数？都是多少？  
    //程序分析：可填在百位、十位、个位的数字都是1、2、3、4。组成所有的排列后再去 掉不满足条件的排列。  
       int geWei,shiWei,baiWei;  
       for(geWei=1;geWei＜5;geWei++) { // 以下为三重循环  
           for(shiWei=1;shiWei＜5;shiWei++) {  
               for (baiWei=1;baiWei＜5;baiWei++) { // 确保geWei、shiWei、baiWei三位互不相同  
                   if (\_\_geWei\_\_\_!=baiWei&&geWei!=shiWei&&shiWei!=\_\_baiWei\_\_\_) {   
                       System.out.println(""+baiWei+\_\_shiWei\_\_\_+geWei);  
                   }  
               }  
           }  
       }  
       }  
      }  
   程序运行控制台效果输出如下  
   123  
   124  
   132  
   134  
   142  
   143  
   213  
   214  
   231  
   234  
   241  
   243  
   312  
   314  
   321  
   324  
   341  
   342  
   412  
   413  
   421  
   423  
   431  
   432

作答结果：✓

正解：

①

②

③

④

解析：

12.public static void main(String[] args) {  
    /\*企业发放的奖金根据利润提成。  
    利润(profit)低于或等于10万元时，奖金可提10%；利润高于10万元，低于20万元时，低于10万元的部分按10%提成，高于10万元的部分，可提成7.5%；  
    20万到40万之间时，高于20万元的部分，可提成5%；40万到60万之间时高于40万元的部分，可提成3%；60万到100万之间时，高于60万元的部分，可提成1.5%；  
    高于100万元时，超过100万元的部分按1%提成。 本月企业 为 1200000.0  求实际发放奖金\*/  
       double profit = 1200000.0;  
       double bonus1,bonus2,bonus4,bonus6,bonus10,bonus = 0;  
       bonus1=100000\*0.1;  
       bonus2=bonus1+100000\*0.075;  
       bonus4=bonus2+200000\*0.05;  
       bonus6=bonus4+200000\*0.03;  
       bonus10=bonus6+400000\*0.015;  
       if(profit＜=100000) {bonus=profit\*0.1;}       
       else if(profit＜=200000) { bonus=\_\_bonus1\_\_\_+(profit-100000)\*0.075;}  
       else if(profit＜=400000) { bonus=bonus2+(profit-200000)\*0.05;}       
       else if(profit＜=600000) { bonus=bonus4+(profit-400000)\*0.03;  }        
       else if(profit＜=1000000) { bonus=bonus6+(profit-\_\_600000\_\_\_)\*0.015; }   
       else if(profit＞1000000) { bonus=bonus10+(profit-1000000)\*0.01;}  
       System.out.println("发放奖金总数为:"+\_\_bonus\_\_\_);  
   }  
   程序运行控制台效果输出如下(注：使用变量)  
   发放奖金总数为:41500.0

作答结果：✓

正解：

①

②

③

解析：

13.public class Run {  
    public static void main(String[] args) {  
    /\*题目：百鸡百钱： 一只公鸡1文钱，一只母鸡2文钱，一只小鸡半文钱，  
                      需要买100只鸡， 正好花完，可以怎么买？有多少种买法？  
    程序分析：  
    穷举：将所有可能的情况都走一遍，用if筛选出符合条件\*/  
    int ff = 0;  
           for(int gj=0;gj＜=\_\_100\_\_\_;gj++){  
               for(int mj=0;mj＜=50;mj++){  
                   for(int xj=0;xj＜=200;xj++){  
                       if(gj+mj+xj==100 && gj+2\*mj+0.5\*xj==100) {  
                           ff++;  
                           System.out.println("可买公鸡"+\_\_gj\_\_\_+"只，母鸡"+mj+"只，小鸡"+\_\_xj\_\_\_+"只");  
                       }  
                   }  
               }  
           }  
           System.out.print("共有"+ff+"种方法购买");  
    }  
   }  
   程序运行控制台效果输出如下  
   …………（省略部分）  
   可买公鸡82只，母鸡6只，小鸡12只  
   可买公鸡85只，母鸡5只，小鸡10只  
   可买公鸡88只，母鸡4只，小鸡8只  
   可买公鸡91只，母鸡3只，小鸡6只  
   可买公鸡94只，母鸡2只，小鸡4只  
   可买公鸡97只，母鸡1只，小鸡2只  
   可买公鸡100只，母鸡0只，小鸡0只  
   共有34种方法购买  
   

作答结果：✓

正解：

①

②

③

解析：

14.public class Run {  
    public static void main(String[] args) {  
    /\*程序分析：求数字0-7所能组成的奇数个数(可能位数为1-8位)，数字可重复，最高位不为0。  
    必须满足要求：  
                   ①最高位不能为0（只能为1-7）。  
                   ②末位数必须为奇数。（末尾数为奇数的数一定为奇数）。  
                   ③除个位/首位，该位可能为8种数字(0-8)。  
                     
            代码逻辑：  
                   分别统计：1位数到8位数的数字可能的奇数个数：  
    ①仅有1位数字时，奇数可能个数为4(１，３、５、７)  
    ②仅有2位数字时，奇数个数为4\*7=28  
    ③当该数为3位数时，奇数个数为：4\*8\*7=224  
    ...  
    ⑧当该数为八位数时，奇数个数为：4\*8\*8\*8\*8\*8\*8\*7  
     
     
    public static void main(String[] args) {  
    int total = 0;  
    //决定数字是几位  
    for(int digit = 1;digit ＜= \_\_8\_\_\_; digit++){  
    if(digit == 1){  
    int jiShuSum = 4;  
    System.out.println(digit + "位数组成的所有奇数可能个数为：" + jiShuSum);  
    total += jiShuSum;  
    }else{  
    int jiShuSum = \_\_4\_\_\_;  
    //「十位」到「最高位-1」  
    for(int k = 0; k ＜ digit - \_\_1\_\_\_; k++){  
    jiShuSum = jiShuSum \* 8;  
    }  
    //最高位  
    jiShuSum = jiShuSum \* 7;  
    System.out.println(digit + "位数组成的所有奇数可能个数为：" + jiShuSum);  
    total +=\_\_jiShuSum\_\_\_;  
    }  
    }  
    System.out.println("奇数的总个数为：" + total);  
    }  
     
     
   控制台输出如下  
   1位数组成的所有奇数可能个数为：4  
   2位数组成的所有奇数可能个数为：28  
   3位数组成的所有奇数可能个数为：224  
   4位数组成的所有奇数可能个数为：1792  
   5位数组成的所有奇数可能个数为：14336  
   6位数组成的所有奇数可能个数为：114688  
   7位数组成的所有奇数可能个数为：917504  
   8位数组成的所有奇数可能个数为：7340032  
   奇数的总个数为：8388608

作答结果：☓

正解：

①

②

③

④

解析：

15.public class Main\_45 {  
    /\*题目：求100 — 1000   所有的水仙花数  
        水仙花数是指一个 n 位数 ( n＞=3 )，它的每个位上的数字的 n 次幂之和等于它本身。  
       （例如：1^3 + 5^3 + 3^3 = 153）  
     
    程序分析：  
    拆分 153   百位 1   十位5  个位 3   
      1\*1\*1 + 5\*5\*5 + 3\*3\*3 = 153  
    \*/  
    public static void main(String[] args) {  
    //定义百位    十位  个位   
     
    int \_\_baiWei\_\_\_,shiWei,geWei = 0;  
    //循环  
    for(int i = 100;i＜\_\_1000\_\_\_;i++){  
    //获取百位数  
    baiWei = i/\_\_100\_\_\_;  
    //获取十位数  
    shiWei = (i%100)/10;  
    //获取个位数  
    geWei = (i%100)%10;  
    // 三位各自的n 次幂之和等于它本身  
    if((baiWei\*baiWei\*baiWei  
      +shiWei\*shiWei\*shiWei+  
      geWei\*geWei\*geWei) ==i){  
    //控制台输出  
    System.out.println(i);  
    }  
    }  
    }  
   }  
   程序运行控制台效果输出如下  
   153  
   370  
   371  
   407

作答结果：✓

正解：

①

②

③

解析：

## JAVASE(05-1.String字符串)

试卷得分：96分

多选题

1.请选择下列正确的字符串定义（ ）

 String str = “123”;  
 String str = ‘123’;  
 String str = new String(“123”);  
 String str = String(“123”);

作答结果：✓

正解：AC

解析：

2.请选择控制台输出结果（ ）  
  public static void main(String [] args){  
         String str = “10”;  
         System.out.println(20+str+false);  
  }

 1020  
 30  
 2010  
 2010false

作答结果：✓

正解：D

解析：

3.请选择控制台输出结果（ ）  
       String str1 = "1";  
       String str2 = "1";  
       String str3 =new String("1");  
       System.out.println(str1.equals(str2));  
       System.out.println(str1.equals(str3));

 弹出true和true  
 弹出true和false  
 弹出false和true  
 弹出false和false

作答结果：✓

正解：A

解析：

4.请选择控制台输出结果（ ）  
       String str1 = "1";  
       String str2 = "1";  
       String str3 =new String("1");  
       System.out.println(str1 == str2);  
       System.out.println(str1 == str3);

 弹出true和true  
 弹出true和false  
 弹出false和true  
 弹出false和false

作答结果：✓

正解：B

解析：

5.请选择控制台输出结果（ ）  
  String str = "abcde";  
  System.out.println(str.charAt(2));

 a  
 b  
 c  
 d

作答结果：✓

正解：C

解析：

6.请选择控制台输出结果（ ）  
  String str = "abcde";  
  System.out.println(str.contains("abcd"));

 cd  
 de  
 true  
 false

作答结果：✓

正解：C

解析：

7.请选择控制台输出结果（ ）  
  String str = "abcde";  
  System.out.println(str.substring(2));

 cde  
 bcd  
 abc  
 abcd

作答结果：✓

正解：A

解析：

8.请选择控制台输出结果（ ）  
  String str = "abcdef";  
  System.out.println(str.substring(2,4));

 cd  
 de  
 cdef  
 abcd

作答结果：✓

正解：A

解析：

9.请选择控制台输出结果（ ）  
  String str = "abcdabcd";  
  System.out.println(str.indexOf("c"));

 3  
 2  
 6  
 5

作答结果：✓

正解：B

解析：

10.请选择控制台输出结果（ ）  
   String str = "abcde";  
   System.out.println(str.replace("cd", "fg"));

 abcde  
 abefg  
 abfge  
 abcdefg

作答结果：✓

正解：C

解析：

11.请选择控制台输出结果（ ）  
   String str = "abcde";  
   System.out.println(str.concat("fg"));

 abcde  
 abefg  
 abfge  
 abcdefg

作答结果：✓

正解：D

解析：

12.请选择控制台输出结果（ ）  
   String str = "abcde";  
   System.out.println(str.startsWith("cd"));

 0  
 1  
 true  
 false

作答结果：✓

正解：D

解析：

13.请选择控制台输出结果（ ）  
   String str = "abcde";  
   System.out.println(str.endsWith("cde"));

 0  
 1  
 true  
 false

作答结果：✓

正解：C

解析：

14.请选择控制台输出结果（ ）  
   String str = "abcd";  
   System.out.println(str.length());

 abc  
 bcd  
 4  
 5

作答结果：✓

正解：C

解析：

15.请选择控制台输出结果（ ）。D  
   String str = "abc";  
   System.out.println(str.toUpperCase());

 abc  
 Abc  
 abC  
 ABC

作答结果：✓

正解：D

解析：

16.请选择控制台输出结果（ ）  
   String str = "       abc            ";  
   System.out.println(str.trim().length());

 0  
 1  
 2  
 3

作答结果：✓

正解：D

解析：

17.请选择控制台输出结果（ ）  
   String str = "   a   b   c   e   f ";  
   while(str.contains(" ")){  
    str = str.replaceAll(" ","");  
   }  
   System.out.println(str.length());

 4  
 5  
 6  
 7

作答结果：✓

正解：B

解析：

18.请选择控制台输出结果（ ）  
   String str = "abcde";  
   System.out.println(str.contains("abcd"));

 cd  
 de  
 true  
 false

作答结果：✓

正解：C

解析：

19.请选择控制台输出结果（ ）  
   String str = "abc";  
   if(str.substring(2,2).isEmpty()){  
    str = str.replaceAll("bc", "d");  
   }  
   System.out.println(str);

 ad  
 abc  
 a  
 d

作答结果：✓

正解：A

解析：

20.请选择控制台输出结果（ ）  
   String str = "abcefg";  
   for(int i = 0;i＜str.substring(2, 5).length();i++){  
    if(str.endsWith("j")){  
    str += "i";  
    }else if(str.contains("k")){  
    str += "j";  
    }else{  
    str += "k";  
    }  
   }  
   System.out.println(str.replace("cefg", "l"));  
   

 ablkji  
 abkjli  
 abklij  
 abjkli

作答结果：✓

正解：A

解析：

21.请补全代码 使该段程序拥有以下功能  
    public static void main(String[] args) {  
    //获取控制台输入  
    Scanner sc = new Scanner(System.in);  
    String str = sc.next();  
    //去空格  
    str = str.\_\_trim\_\_\_();  
    //反转  
    char [] arr =str.toCharArray();  
    for(int i =\_\_0\_\_\_;i＜arr.length/2;i++){  
    //交换变量实现倒序  
    char temp = arr[i];  
    arr[i] = arr[arr.length-1-i];  
    arr[arr.length-1-i] = \_\_temp\_\_\_;  
    }  
    str = new String(arr);  
    System.out.println(str);  
    }  
     
       1.获取控制台输入字符串  
       2.去掉该字符串两端空白字符串  
       3.使该字符串倒序输出  
       运行例：  
       abcde  
       edcba  
         
       1234  
       4321

作答结果：✓

正解：

①

②

③

解析：

22.public class Run {  
    public static void main(String[] args) {  
    /\*题目：练习String类常用方法-1。\*/  
    String str = "abcdabcdefg";  
    //需求1  打印该字符串长度  
    System.out.println(str.\_\_length()\_\_\_);  
    //需求2 根据索引（下标）访问字符串中元素 打印第三个元素 =＞'c'  
    System.out.println(str.charAt(2));  
    //需求3 用concat方法拼接字符串 =＞ 拼接"hi"后并打印  
    System.out.println(str.\_\_concat\_\_\_("hi"));  
    //需求4 判断该字符串是否以"h"结尾，根据该结果打印 "是" 或者 "不是"  
    if(str.\_\_endsWith\_\_\_("h")){  
    System.out.println("是");  
    }else{  
    System.out.println("不是");  
    }  
    //需求5 输出字符"d" 第一次出现所在Index(下标) 和最后一次出现所在Index(下标)  
    System.out.println("d第一次出现在"+str.indexOf("d"));  
    System.out.println("d最后一次出现在"+str.lastIndexOf("d"));  
    }  
   }  
   请补全代码，使得程序运行控制台输出如下  
   11  
   c  
   abcdabcdefghi  
   不是  
   d第一次出现在3  
   d最后一次出现在7

作答结果：✓

正解：

①

②

③

解析：

23.public class Run {  
    /\*题目：练习String类常用方法-2。\*/  
    public static void main(String[] args) {  
    String str = "zhangsan";  
    //需求1 判断该字符串是否包含"zh"，根据该结果打印 "包含" 或者 "不包含"  
    if(str.\_\_contains\_\_\_("zh")){  
    System.out.println("包含");  
    }else{  
    System.out.println("不包含");  
    }  
    //需求2  请将该字符串中所有"s" 替换为 "l" ，并且打印出来  
    System.out.println(str.\_\_replace\_\_\_("s", "l"));  
    //需求3   截取该字符串 从第3位起 到结束(包括第3位)， 并且打印出来  
    System.out.println(str.substring(2));  
    //需求4   截取该字符串 从第4位起 到7位(包括第4位,不包括第7位)， 并且打印出来  
    System.out.println(str.substring(3, \_\_6\_\_\_));  
    //需求5 将该段字符串转换成大写，并且打印出来  
    System.out.println(str.\_\_toUpperCase\_\_\_());  
    }  
   }  
   请补全代码，使得程序运行控制台输出如下  
   包含  
   zhanglan  
   angsan  
   ngs  
   ZHANGSAN  
   

作答结果：✓

正解：

①

②

③

④

解析：

24.public class Run {  
    /\*  
     \* 题目：寻找两个字符串的最大公共子字符串。  
     \*   
     \* 程序分析： 1.判断较长字符串中是否包含较短字符串，如果包含，则较短字符串则为最大公共子串。  
     \* 2.如果不包含，就对较短字符串以长度递减的方式取子串，去较长字符串中判断是否包含，如果包含就找到了，不用再找了。  
     \* 3.重点：对字符串以长度递减的方式取子串  
     \*/  
    public static void main(String[] args) {  
    String str1 = "abcdefghijk";  
    String str2 = "cdefghij";  
    String maxCommon = getMaxCommon(str1,str2);  
    if(maxCommon != null){  
    System.out.println(str1+"和"+str2+"最大公共子字符串为:");  
    System.out.println(\_\_maxCommon\_\_\_);  
    }  
    }  
     
    public static String getMaxCommon(String str1,String str2){  
    //断定较长字符串和较短字符串  
    String max=(str1.length()＞str2.length())?\_\_str1\_\_\_:str2;  
    String min=(str1.equals(max))?str2:str1;  
    //按长度递减的方式截取子串，从min.length~~1  
    for(int i=min.length();i＞=\_\_1\_\_\_;i--){  
    for(int x=0,y=x+i;y＜=min.length();x++,y++){  
    String childStr=min.\_\_substring\_\_\_(x,y);  
    //若较长字符串中包含此子串，则找到了  
    //否则继续找  
    if(max.contains(childStr)){  
    return childStr;  
    }  
    }  
    }  
    return null;//如果没有返回 null  
    }  
   }  
   请补全代码，使得程序运行控制台输出如下  
   abcdefghijk和cdefghij最大公共子字符串为:  
   cdefghij

作答结果：☓

正解：

①

②

③

④

解析：

25.public class Run {  
     
    public static void main(String[] args) {  
    /\*题目：将一个字符串逆序输出。  
     
    程序分析：用第一个与最后一个交换。\*/  
    String str = "123";  
     
    System.out.println(\_\_reverse\_\_\_("hello"));  
     
    }  
    public static String reverse(String \_\_input\_\_\_){  
        // 将字符串转换为数组  
        char [] inputCharArr = input.toCharArray();  
        // 获取字符串长度  
        int len = inputCharArr.length;  
        // 创建同等长度数组  
        char [] outCharArr = new \_\_char\_\_\_[len];  
        // 交换顺序  
    for(int i =0;i＜len;i++){  
    outCharArr[i] = inputCharArr[len-1-i];  
        }  
    // 返回交换顺序后的新字符串  
    String output = new String(outCharArr);  
    return \_\_output\_\_\_;   
    }  
   }  
   请补全代码，使得程序运行控制台输出如下  
   olleh

作答结果：✓

正解：

①

②

③

④

解析：

26.public class Run {  
    public static void main(String[] args) {  
    /\*题目：判断两个字符串(未知长度)是否相似  (打乱顺序按任意顺序排列重组后相同)  
     \*  
              例1：abc和 acb 和 cab相似   
           abc 和 aab不相似 ，  abc 和 acc不相似   
              例2：abca 和 aabc 和baac 相似  
           abca 和 abbc不相似 ， abca 和 abcc不相似   
     
    程序分析       
           1.两个字符串，任意顺序打乱，无法用任何string类自带方法实现。  
           2.前提条件：字符串长度不相等则不可能相似(任意顺序重组后长度也不相等)。  
         3.两个字符串分别转为char []。  
         4.遍历各自的char[] 将所有char 转换为int  
         5.比较各自所有元素转换为int之后的和可判断是否相似  。\*/  
     
    System.out.println(isSimilar("abc","cba"));  
     
    }  
    public static boolean isSimilar(String str1,String str2){  
    //字符串长度比较  
    if(str1.length()== str2.length()){  
    //String 转换为char []  
    char [] arr1 = str1.toCharArray();  
    char [] arr2 = str2.toCharArray();  
    // 各自的和  
    int sum1 =0;  
    int \_\_sum2\_\_\_ =0;  
    // 遍历 将所有char 转换为int  
    for(char c :arr1){  
    //char 类型转换为int 类型   
    sum1 +=(int )c;  
    }  
    for(char c :arr2){  
    //char 类型转换为int 类型  
    sum2 +=(int )c;  
    }  
    if(sum1 == sum2){  
    // 相等  
    return \_\_true\_\_\_;  
    }else{  
    // 不相等  
    return false;  
    }  
    }else{  
    //字符串长度不相等则不相似  
    return \_\_false\_\_\_;  
    }  
    }  
   }  
   请补全代码，使得程序运行控制台输出如下  
   true  
     
     
   

作答结果：✓

正解：

①

②

③

解析：

## JAVASE(07-1.数组及冒泡排序练习题)

试卷得分：0分

多选题

1.请补全代码，使控制台输出如下   
  package run;  
  public class Main  {  
   public static void main(String[] args) {  
   int [] arr = {5,1,3,9,7};  
   System.out.println("排序前:");  
   for(int \_\_e\_\_\_ :arr){  
   System.out.print(e);  
   }  
   System.out.println();  
   //冒泡排序  
   for(int i = \_\_0\_\_\_;i＜\_\_arr.length-1\_\_\_;i++){  
   for(int j = 0;j＜\_\_arr.length-1-i\_\_\_;j++){  
   if(arr[j]＞arr[j+1]){  
   int temp = arr[j];  
   arr[j] = arr[j+1];  
   arr[j+1] = temp;  
   }  
   }  
   }  
   System.out.println("排序后:");  
   for(int e :arr){  
   System.out.print(e);  
   }  
   }  
  }  
    
  控制台输出如下  
  排序前:  
  51397  
  排序后:  
  13579

作答结果：☓

正解：

①

②

③

④

解析：

## JAVASE(08-1.方法(函数)的介绍)

试卷得分：94分

多选题

1.下列关于说法正确的是（ ）

 C语言，VB等开发语言是面向过程的语言  
 C语言，VB等开发语言是面向对象的语言  
 JAVA，C#等开发语言 面向过程的语言  
 JAVA，C#等开发语言 面向对象的语言

作答结果：✓

正解：AD

解析：

2. 下列说法正确的是（ ）

 面向过程=＞ 函数（かんすう）  
 面向过程=＞ method（メソード）  
 面向对象=＞ 函数（かんすう）  
 面向对象=＞ method（メソード）

作答结果：✓

正解：AD

解析：

3.下列说法正确的是（ ）

 JAVA中方法定义在类外面  
 JAVA中方法定义在类里面  
 方法中可以调用其他方法  
 方法中不可以调用其他方法

作答结果：✓

正解：BC

解析：

4.下列关于JAVA 方法定义说法正确的是（ ）

 除了匿名函数，函数必须拥有函数名  
 函数一定没有函数名  
 方法可以没有参数  
 方法必须拥有参数

作答结果：✓

正解：AC

解析：

5.下列关于JAVA 方法定义说法正确的是（ ）

 方法头中可以不用声明具体的返回类型  
 方法头中可以必须声明具体的返回类型  
 方法可以没有方法体  
 方法必须拥有方法体

作答结果：☓

正解：BC

解析：

6.关于方法重载正确的是（ ）

 发生子类和父类之间  
 发在一个类中  
 构造方法支持重载  
 构造方法不支持重载

作答结果：✓

正解：BC

解析：

7.重载的定义是（ ）

 方法名相同，相同的参数个数  
 方法名不同，不同的参数类型  
 方法名相同，不同的参数类型  
 方法名不同，相同的参数类型

作答结果：✓

正解：C

解析：

8.下列关于JAVA方法定义错误的是（ ）

 方法参数必须拥有类型  
 方法参数必须可以没有类型  
 方法中参数只能有0个或者1个  
 方法中参数只能有0个或者多个(无上限)

作答结果：✓

正解：BC

解析：

9.下列关于JAVA方法定义错误的是（ ）

 方法的声明不能嵌套，方法中不能在声明方法  
 方法的声明可以嵌套，方法中可以继续声明方法  
 同名方法可能有多个(可重载多次)  
 同名方法只能有一个(只可重载一次)

作答结果：✓

正解：BD

解析：

10.请选择正确的一项()  
    method1("100");  
    }  
    public static void method1(int num){  
    System.out.println(num);  
    }  
   }

 编译错误  
 运行错误  
 运行控制台输出100  
 运行控制台输出0

作答结果：✓

正解：A

解析：

11.请选择正确的一项()  
   package run;  
     
   public class Main  {  
    public static void main(String[] args) {  
    method1(10);  
    }  
    public static void method1(){  
    System.out.println("20");  
    }  
    public static void method1(int num){  
    System.out.println(num);  
    }  
   }

 编译错误  
 运行错误  
 运行控制台输出20  
 运行控制台输出10

作答结果：✓

正解：D

解析：

12.请选择正确的一项()  
   package run;  
   public class Main  {  
    public static void main(String[] args) {  
    method1("10",20);  
    }  
    public static void method1(int num,String str){  
    System.out.println(num+str);  
    }  
    public static void method1(String str,int num){  
    System.out.println(str+num);  
    }  
   }

 编译错误  
 编译错误  
 运行控制台输出1020  
 运行控制台输出2010

作答结果：✓

正解：C

解析：

13.请选择正确的一项()  
   package run;  
   public class Main  {  
    public static void main(String[] args) {  
    int result = add(10,20);  
    System.out.println(result);  
    }  
    public static void add(int num1,int num2){  
    int result=num1 + num2;  
    }  
   }

 编译错误  
 运行错误  
 运行控制台输出30  
 运行控制台输出1020

作答结果：✓

正解：A

解析：

14.请选择正确的一项()  
   package run;  
     
   public class Main  {  
    public static void main(String[] args) {  
    int a =  10;  
    System.out.println(method1(a));  
    }  
    public static int method1(int a){  
    return a\*a;  
    }  
    public static String method1(String a){  
    return a+a;  
    }  
   }

 编译错误  
 运行错误  
 运行控制台输出100  
 运行控制台输出1010

作答结果：✓

正解：C

解析：

15.请选择正确的一项()  
   package run;  
   public class Main  {  
    public static void main(String[] args) {  
    int a = 20;  
    String b = "10";  
    System.out.println(method1(b,method1(a,b)));;  
    }  
    public static int method1(int a,String b){  
    return new Integer(a+b);  
    }  
    public static String method1(String a,int b){  
    return b+a;  
    }  
   }

 运行控制台输出102010  
 运行控制台输出 201010  
 运行控制台输出 102020  
 运行控制台输出202010

作答结果：✓

正解：B

解析：

16.package run;  
     
   public class Run{  
    /\*题目：方法定义以及调用基础练习-1  
    \*/  
     
    public static void main(String [] args){  
    String str = "abc";  
    int num = 10;  
    //需求1 定义方法method,该方法接受1个int类型参数，变量名为num  
    //     方法体中控制台输出num的平方,无返回值  
    //     调用method方法 传入变量num  
    method(num);  
    //需求2 定义方法method1,该方法接受1个String类型参数，变量名为str  
    //     方法体中控制台输出str首个字符和末尾字符拼接,无返回值  
    //     控制台输出例："首字母:a 末尾字母:c"    
    //     调用method方法 传入变量str  
    method1(\_\_str\_\_\_);  
    //需求3 定义方法method2,该方法接受2个参数，第一个int类型，第二个String类型，  
    //     返回两参数相加的结果(int类型在前)  
    //     定义变量String类型 result   
    //     调用method2方法 传入变量num,str并将返回值赋给result  
    //     控制台输出result  
    String result = method2(num,str);  
    System.out.println(result);  
    }  
    public static void method(\_\_int\_\_\_ \_\_num\_\_\_){  
    System.out.println(num\*num);  
    }  
    public static void method1(String str){  
    System.out.println("首字母:"+str.charAt(0)+" 末尾字母:"+str.charAt(str.\_\_length\_\_\_()-1));  
    }  
    public static String method2(int num,String str){  
    return num + str;  
    }  
   }  
   请补全代码，使得运行程序，控制台输出效果如下  
   100  
   首字母:a 末尾字母:c  
   10abc

作答结果：✓

正解：

①

②

③

④

解析：

17.package run;  
     
   public class Run{  
    /\*题目：编写方法，求数组元素的平均值。  
    需求：  
    1、定义一个方法，求数组的平均值  
    2、在主方法中定义变量，并初始化一个int类型的数组，调用方法求数组元素的平均值，  
    并将平均值打印输出  
    \*/  
    public static void main(String[] args) {  
    int [] arr = {5,12,8,6,9};  
    getAvg(arr);  
    }  
    public static void getAvg(\_\_int\_\_\_ [] array){  
    int \_\_sum\_\_\_ = 0;  
    for(int i = 0;i＜array.\_\_length\_\_\_;i++){  
    sum \_\_+=\_\_\_ array[i];  
    }  
    System.out.println(sum/array.length);  
    }  
   }  
   请补全代码，使得运行程序，控制台输出效果如下  
   8

作答结果：✓

正解：

①

②

③

④

解析：

18.package run;  
     
   public class Run {  
    /\*题目：重载方法，控制台输出数组从指定索引到指定索引中的元素。  
    需求：  
    1、定义一个方法，方法名为myPrint，一个参数类型为String []  
           方法体中遍历传入数组并在控制台输出所有元素  
    2、重载myPrint，变为三参，第一个参数为String []，  
           第二个，第三个参数都为int 类型 ，分别代表起始，和结束位置索引  
           方法体中   
      2.1判断第二，第三参数是否为合法索引(起始索引＞=0 ，起始＜=结束,结束＜=数组长度-1)  
      2.2 合法的情况 遍历传入数组并在控制台输出数组从起始索引到结束索引中的元素  
      2.3 不合法的情况    控制台输出 "请传入正确的索引！"      
        3、调用三参myPrint  
     \*/  
    public static void main(String[] args) {  
    String [] arr = {"a","b","c","d","e","f"};  
    myPrint(\_\_arr\_\_\_,1,3);  
    }  
    //遍历输出所有元素  
    public static void \_\_myPrint\_\_\_(String [] arr){  
    for(int i = 0;i＜arr.length;i++){  
    System.out.println(arr[i]);  
    }  
    }  
    //输出指定索引位置元素  
    public static void myPrint(String [] arr,int \_\_startIndex\_\_\_,int lastIndex){  
    //合法索引判断  
    if(0＜=startIndex && startIndex＜= lastIndex && lastIndex＜=arr.length-\_\_1\_\_\_){  
    for(int i = startIndex;i＜=lastIndex;i++){  
    System.out.println(arr[i]);  
    }  
    }else{  
    System.out.println("请传入正确的索引！");  
    }  
    }  
   }  
   请补全代码，使得运行程序，控制台输出效果如下  
   b  
   c  
   d

作答结果：✓

正解：

①

②

③

④

解析：

19.package run;  
     
   public class Run {  
    /\*题目：控制台输出倒序99乘法表。  
    需求：  
    1、定义一个方法，方法名为myPrint，方法体中控制台输出倒序乘法表     
       2、调用方法myPrint  
    \*/  
    public static void main(String[] args) {  
    \_\_myPrint\_\_\_();  
    }  
     
    public \_\_static\_\_\_ void myPrint(){  
    for(int i = \_\_9\_\_\_;i＞0;i--){  
    for(int j = 1;j＜=i;j++){  
    System.out.print(j+"\*"+i+" = "+i\*j+" ");  
    }  
    System.out.\_\_println\_\_\_();  
    }  
    }  
   }  
   请补全代码，使得运行程序，控制台输出效果如下  
   1\*9 = 9 2\*9 = 18 3\*9 = 27 4\*9 = 36 5\*9 = 45 6\*9 = 54 7\*9 = 63 8\*9 = 72 9\*9 = 81   
   1\*8 = 8 2\*8 = 16 3\*8 = 24 4\*8 = 32 5\*8 = 40 6\*8 = 48 7\*8 = 56 8\*8 = 64   
   1\*7 = 7 2\*7 = 14 3\*7 = 21 4\*7 = 28 5\*7 = 35 6\*7 = 42 7\*7 = 49   
   1\*6 = 6 2\*6 = 12 3\*6 = 18 4\*6 = 24 5\*6 = 30 6\*6 = 36   
   1\*5 = 5 2\*5 = 10 3\*5 = 15 4\*5 = 20 5\*5 = 25   
   1\*4 = 4 2\*4 = 8 3\*4 = 12 4\*4 = 16   
   1\*3 = 3 2\*3 = 6 3\*3 = 9   
   1\*2 = 2 2\*2 = 4   
   1\*1 = 1 

作答结果：✓

正解：

①

②

③

④

解析：

## AVASE(09-1.类和对象)

试卷得分：100分

多选题

1.下列说法正确的是（ ）

 类是抽象的  
 对象是抽象的  
 类是具体的  
 对象是具体的

作答结果：✓

正解：AD

解析：

2.下列说法正确的是（ ）

 在JAVA中,把具有相同属性,方法的事物叫做类  
 在JAVA中,把具有相同属性,方法的事物叫做对象  
 类名首字母必须大写  
 类名首字母必须小写

作答结果：✓

正解：AC

解析：

3.在同一个类的定义中可以有两个同名函数，这种现象称为函数（ ）

 封装  
 继承  
 覆盖  
 重载

作答结果：✓

正解：D

解析：

4.下列说法正确的是（ ）

 类是对象的抽象  
 对象是类的抽象  
 类是对象的具体实例  
 对象是类的具体实例

作答结果：✓

正解：AD

解析：

5.请选择下列选项中定义正确的类（ ）

 class Person{ int age; String name; }  
 class person{ int age; String name; }  
 Person class { int age; String name; }  
 person class { int age; String name; }

作答结果：✓

正解：A

解析：

6.下列关于对象说法正确的是（ ）

 Student student = new Student() ；=＞声明对象与创建对象  
 Student student ； =＞创建对象  
 对象名首字母大写  
 对象名首字母小写

作答结果：✓

正解：AD

解析：

7.下列关于包说法正确的是（ ）

 包用来存放类  
 包用来存放对象  
 除了默认包，需在第一行使用package关键字指定当前类在哪一个包  
 可在任意行使用package关键字指定当前类在哪一个包

作答结果：✓

正解：AC

解析：

8.下列关于包说法正确的是（ ）

 同一包下面的类，需要导入后才能使用  
 同一包下面的类，不需要导入可以直接使用  
 import \* 指导入某包下面所有类  
 import \* 指导入某包下面第一个类

作答结果：✓

正解：BC

解析：

9.下列关于包说法正确的是（ ）

 package关键字放在import关键字前面  
 package关键字放在import关键字后面  
 package关键字和import关键字没有顺序，先后均可  
 jdk默认帮我们导入java.lang包

作答结果：✓

正解：AD

解析：

10.为了使包 ch4在当前程序中可见，可以使用的语句是（ ）

 import ch4.\*;  
 package ch4.\*;  
 ch4 import;  
 ch4 package;

作答结果：✓

正解：A

解析：

11.请选择下列说法中正确的()

 不需要定义类，就能创建对象  
 对象中必须有属性和方法  
 .属性可以是简单变量，也可以是一个对象  
 属性必须是简单变量

作答结果：✓

正解：C

解析：

12.请选择下列关于main方法 说法中正确的()

 方法 main（）只能放在公共类中  
 main()的头定义可以根据情况任意更改  
 一个类中可以没有 main()方法  
 所有对象的创建都必须放在 main()方法中

作答结果：✓

正解：C

解析：

13.public class Test {  
       public static void main() {  
             System.out.print("aaa");  
       }  
    }  
   下面说法正确的是?

 程序无法通过编译，因为main方法声明方式不对  
 程序可以通过编译，但运行程序行，会提示没有找到main方法的异常  
 程序正常运行，并输出aaa  
 程序正常运行，并输出aaa

作答结果：✓

正解：B

解析：

14.请选择下列说法中正确的()

 null指尚未被创建的对象  
 null指的是空字符串  
 比较字符串时，使用equals方法  
 比较字符串时，使用==

作答结果：✓

正解：AC

解析：

15.请选择下列关于编译器提示说法中正确的()

 黄色=＞警告，代码可以运行  
 黄色=＞编译错误，代码无法运行  
 红色=＞警告，代码可以运行  
 红色=＞编译错误，代码无法运行

作答结果：✓

正解：AD

解析：

16.package run;  
     
   public class Run {  
    /\*题目：类与对象基础练习-1。  
     定义类Person，该类拥有两个成员属性 ,一个成员方法  
    分别是  属性名age类型为int，属性名name类型为String  
    方法名为walk，无参数无返回值，方法体中输出 "我在散步"  
    在Run类中，定义main方法 ，创建Person类对象person  
    调用walk成员方法  
    程序分析：  
    \*/  
    public static void main(String[] args) {  
    Person person = new Person();  
    person.walk();  
    }  
   }  
   \_\_class\_\_\_ Person{  
    public int age;  
    public String \_\_name\_\_\_;  
    public \_\_void\_\_\_ walk(){  
    System.out.println("我在散步");  
    }  
   }  
     
   请补全代码，使得运行程序，控制台输出效果如下  
   我在散步

作答结果：✓

正解：

①

②

③

解析：

17.package run;  
     
   public class Run {  
    /\*题目：类与对象基础练习-2。  
     定义类Person，该类拥有3个成员属性 ,一个成员方法  
    分别是  属性名age类型为int，属性名name类型为String  
    属性名 height类型为double  
    方法名为sayHi，无参数无返回值，方法体中输出 成员属性和字符串的拼接  
    例=＞"你好，我是jack,我今年18岁，我身高180.0"  
    在Run类中，定义main方法 ，创建Person类对象person  
    给person三个属性赋值，调用sayHi成员方法  
    程序分析：  
    \*/  
    public static void main(String[] args) {  
    Person person = new Person();  
    person.age = 18;  
    person.name = "jack";  
    person.height = 180;  
    person.\_\_sayHi\_\_\_();  
    }  
   }  
   class \_\_Person\_\_\_{  
    public int age;  
    public String name;  
    public double height;  
    public void sayHi(){  
    System.out.println("你好，我是"+this.name+",我今年"+  
    this.age+"岁"+"\_\_我身高\_\_\_"+this.height);  
    }  
   }  
   请补全代码，使得运行程序，控制台输出效果如下  
   你好，我是jack,我今年18岁我身高180.0

作答结果：✓

正解：

①

②

③

解析：

18.package run;  
     
   public class Run {  
    /\*题目：类与对象基础练习-3。  
     定义类Dog,该类有两个属性   
    属性名age类型为int,属性名name类型为String  
     定义类Person,该类拥有3个成员属性,一个成员方法  
    属性名age类型为int,属性名name类型为String,属性名pet类型为Dog  
    方法名位walkWithPet，方法体中打印 实例属性 name 拼接字符串 " walk with "  
       再拼接实例属性 pet.name   
    在Run类中，定义main方法 ，在方法体中  
    创建Dog类对象dog，给dog对象两个属性赋值  
    创建Person类对象person,给person三个属性赋值  
    调用walkWithPet方法  
    程序分析：  
    \*/  
    public static void main(String[] args) {  
    Dog dog = new Dog();  
    dog.age = 1;  
    dog.name = "xiaoHuang";  
    Person person = new Person();  
    person.age = 18;  
    person.name = "jack";  
    person.pet = \_\_dog\_\_\_;  
    person.walkWithPet();  
    }  
   }  
   class Person{  
    public int \_\_age\_\_\_;  
    public String name;  
    public Dog pet = new Dog();  
    public void walkWithPet(){  
    System.out.println(this.name+" walk with "+this.\_\_pet\_\_\_.name);  
    }  
   }  
   class Dog{  
    public int age;  
    public \_\_String\_\_\_ name;  
   }  
   请补全代码，使得运行程序，控制台输出效果如下  
   jack walk with xiaoHuang

作答结果：✓

正解：

①

②

③

④

解析：

## JAVASE(10-1.八个基本类型的包装类)

试卷得分：92分

多选题

1.下列说法正确的是（ ）

 装箱就是自动将包装器类型转换为基本数据类型  
 装箱就是自动将基本数据类型转换为包装类型  
 拆箱就是自动将包装器类型转换为基本数据类型  
 拆箱就是自动将基本数据类型转换为包装类型

作答结果：✓

正解：BC

解析：

2.下列说法正确的是（ ）

 Integer total = 99; 自动装箱  
 Integer total = 99; 自动拆箱  
 int totalprim = total; 自动装箱  
 int totalprim = total; 自动拆箱

作答结果：✓

正解：AD

解析：

3. 请选择正确的一项()  
  public static void main(String[] args) {  
   Integer i1 = 100;  
   Integer i2 = 100;  
   System.out.println(i1==i2);    
  }

 编译错误  
 编译错误  
 控制台输出true  
 控制台输出false

作答结果：✓

正解：C

解析：

4.请选择正确的一项()  
  public static void main(String[] args) {  
   Double i1 = 100.0;  
   Double i2 = 100.0;  
   System.out.println(i1==i2);    
  }

 编译错误  
 运行错误  
 控制台输出true  
 控制台输出false

作答结果：✓

正解：D

解析：

5.请选择正确的一项()  
  public static void main(String[] args) {  
   Integer i1 =  130;  
   Integer i2 =  130;  
   System.out.println(i1==i2);    
  }

 编译错误  
 编译错误  
 控制台输出true  
 控制台输出false

作答结果：☓

正解：D

解析：

6.请选择正确的一项()  
  public static void main(String[] args) {  
   Integer i1 = 130;  
   Double  i2 =  130.0;  
   System.out.println(i1==i2);    
  }

 编译错误  
 运行错误  
 控制台输出true  
 控制台输出false

作答结果：✓

正解：A

解析：

7.请选择正确的一项()  
  public static void main(String[] args) {  
   Integer i1 = 130;  
   Double  i2 =  130.0;  
   System.out.println(i1.equals(i2));    
  }

 编译错误  
 运行错误  
 控制台输出true  
 控制台输出false

作答结果：✓

正解：D

解析：

8.请选择正确的一项()  
  public static void main(String[] args) {  
   Integer  i1 =   new Integer(10);  
   Integer  i2 =   new Integer(10);  
   System.out.println(i1 == i2);    
  }

 编译错误  
 运行错误  
 控制台输出true  
 控制台输出false

作答结果：✓

正解：D

解析：

9.请选择正确的一项()  
  public static void main(String[] args) {  
   Integer i1 = 10;  
   Integer i2 = Integer.valueOf(10);  
   Integer i3 = new Integer(10);  
          System.out.println(i1 == i2);    
   System.out.println(i2 == i3);    
  }

 控制台打印true和true  
 控制台打印false和false  
 控制台打印true和false  
 控制台打印false和true

作答结果：✓

正解：C

解析：

10.选择不是 Java语言中的基本数据类型的选项()  
      

 byte　  
 long　  
 String　  
 char

作答结果：✓

正解：C

解析：

11.package run;  
     
   public class Run{  
    /\*题目：8个基本类型及包装类基础练习-1  
    \*/  
    public static void main(String[] args) {  
    int i = 20;  
    //将int类型转换为包装类型  
    Integer wrapperi = \_\_new\_\_\_ Integer(i);  
    //将包装类型转换为基本类型  
    int j = wrapperi.intValue();  
    //将Integer 转换为String类型  
    String str = wrapperi.\_\_toString\_\_\_();  
    //将String 转换为Integer类型  
    Integer k = Integer.valueOf(str);  
    //将Integer 转换为Double类型  
    Double d = wrapperi.\_\_doubleValue\_\_\_();  
    }  
   }  
   请根据注释补全代码

作答结果：✓

正解：

①

②

③

解析：

12.package run;  
     
   public class Run {  
     
    /\*题目：8个基本类型及包装类基础练习-2  
    需求:  
    1.定义静态方法getInteger,传入一参int类型，方法体返回对应Integer包装类型对象  
    2.定义静态方法getInt,传入一参Integer包装类型对象类型，方法体返回对应int值  
    3.定义静态方法getCharacter,传入一参char类型，方法体返回对应Character包装类型对象  
    4.定义静态方法getChar,传入一参Character包装类型对象类型，方法体返回对应char值  
    \*/       
       public static Integer getInteger(\_\_int\_\_\_ i){  
        return new Integer(i);  
       }  
       public static int getInt(Integer wrapper){  
        return wrapper.intValue();  
       }  
       public static Character getCharacter(char c){  
        return new Character(\_\_c\_\_\_);  
       }  
       public static char getChar(\_\_Character\_\_\_ wrapper){  
        \_\_return\_\_\_ wrapper.charValue();  
       }  
   }  
   请根据注释补全代码

作答结果：✓

正解：

①

②

③

④

解析：

13.package run;  
     
   public class Run {  
    /\*题目：8个基本类型及包装类基础练习-3  
    需求:  
    1.定义静态方法getLongObject,传入一参long类型，方法体返回对应Long包装类型对象  
    2.定义静态方法getLong,传入一参Long包装类型对象类型，方法体返回对应int值  
    3.定义静态方法getBooleanObject,传入一参boolean类型，方法体返回对应Boolean包装类型对象  
    4.定义静态方法getBoolean,传入一参Boolean包装类型对象类型，方法体返回对应boolean值  
    \*/       
       public static Long getLongObject(long \_\_l\_\_\_){  
        return new Long(l);  
       }  
       public static long getLong(\_\_Long\_\_\_ wrapper){  
        return wrapper.longValue();  
       }  
       public static Boolean getBooleanObject(booleanb){  
        return \_\_new\_\_\_ Boolean(b);  
       }  
       public static boolean getBoolean(Boolean wrapper){  
        return wrapper.booleanValue();  
       }  
   }  
   请根据注释补全代码

作答结果：✓

正解：

①

②

③

解析：

## JAVASE(11-1.类的继承)

试卷得分：75分

多选题

1.下列说法正确的是（ ）

 JAVA中类的继承是单继承，一个子类只能继承一个父类  
 JAVA中类的继承是多继承，一个子类可以继承多个父类  
 JAVA中类使用extends继承另一个类  
 JAVA中类使用implements继承另一个类

作答结果：✓

正解：AC

解析：

2.下列说法正确的是（ ）

 当一个类没有继承关键字，则默认继承Object  
 子类继承父类非private的属性，方法  
 子类继承父类所有的属性，方法  
 继承降低了类之间的耦合性

作答结果：✓

正解：AB

解析：

3.请选择下列说法中正确的()

 继承减少了代码冗余  
 继承实现了代码的复用  
 OOA=＞面向对象分析  
 OOA=＞面向对象设计

作答结果：✓

正解：ABC

解析：

4.请选择下列说法中正确的()

 OOD=＞面向对象设计  
 OOD=＞面向对象分析  
 OOP =＞面向对象编程  
 OOP =＞面向对象设计

作答结果：✓

正解：AC

解析：

5.请选择面向对象语言的三大特性()

 封装  
 继承  
 多态  
 跨平台

作答结果：✓

正解：ABC

解析：

6.请选择下列中类的继承定义正确的选项()

 public class Child Extends Huamn{}  
 class public Child Extends Huamn{}  
 class public Child Extends Huamn{}  
 public class Child extends Huamn{}

作答结果：✓

正解：D

解析：

7.请选择下列中关于继承的特点说法正确的()

 子类可以拥有父类的属性和方法  
 子类可以拥有自己的属性和方法  
 可以声明父类，创建子类  
 可以声明子类，创建父类

作答结果：☓

正解：ABC

解析：

8.请选择下列中关于继承的特点说法正确的()

 创建什么类型，就能强转为什么类型  
 子类可以拥有自己的属性和方法  
 方法中参数声明子类时，可以传入父类对象  
 方法中参数声明父类时，可以传入子类对象

作答结果：☓

正解：ABD

解析：

9.请选择下列说法中错误的 ()

 super关键字=＞访问父类的属性和方法  
 @Override注释=＞对重写代码进行检测注释  
 this.关键字 =＞访问父类的属性和方法  
 super关键字必须位于构造方法第一行

作答结果：☓

正解：C

解析：

10.请选择正确的一项()  
   public class SuperType {  
    public void method(){  
    System.out.println("111");  
    }  
   }  
     
   public class SubType extends SuperType{  
    @Override  
    public void method(){  
    System.out.println("222");  
    }  
   }  
   public class Main {  
     
    public static void main(String[] args) {  
    SubType st = new SubType();  
    st.method();  
    }  
     
   }

 编译错误  
 运行错误  
 控制台输出111  
 控制台输出222

作答结果：✓

正解：D

解析：

11.package run;  
     
   class \_\_SuperType\_\_\_ {  
    public int type = 1;  
    private String msg = "hello";  
    public void sayHi2 (){  
    System.out.println("niHao");  
    }  
    private void sayHi1 (){  
    System.out.println("niHao");  
    }  
   }  
     
   class SubType \_\_extends\_\_\_ SuperType{  
   }  
     
   public class Run{  
    /\*题目：类的继承-1。  
     定义类SuperType,该类有两个属性   
    属性名type类型为int默认值为1,属性名msg类型为String默认值为"hello"  
    定义类SubType,该类继承SuperType  
       
    在Run类中，定义main方法 ，在方法体中  
    创建SubType类对象subType  
    控制台输出subType对象的从父类(SubType)继承的属性  
    调用subType对象的从父类(SubType)继承的方法，使得控制台输出 niHao  
    程序分析：  
    \*/  
    public static void main(String[] args) {  
    SubType subType = \_\_new\_\_\_ SubType();  
    System.out.println(subType.type);  
    subType.sayHi2();  
    }  
   }

作答结果：✓

正解：

①

②

③

解析：

12.package run;  
     
   class Animal {  
    public \_\_String\_\_\_ name;  
    public void eat(){  
    System.out.println(this.name+" eat");  
    }  
   }  
   class Person extends Animal{  
    public void eat(){  
    System.out.println(this.name+"eatfood");  
    }  
   }  
   class Tiger \_\_extends\_\_\_ Animal{  
    public void eat(){  
    System.out.println(this.name+"eat\_\_meat\_\_\_");  
    }  
   }  
   public class Run{  
    /\*题目：类的继承-2。  
     定义类Animal,该类有一个属性 ，一个方法  
                属性名name类型为String  
    方法名eat，无参无返回值，方法体中打印 this.name拼接" eat"  
    定义类Person,该类继承Animal  
    覆盖(重写)方法eat,方法体中打印 this.name拼接" eat food"  
    定义类Tiger,该类继承Animal  
    覆盖(重写)方法eat,方法体中打印 this.name拼接" eat meat"  
       
    在Run类中，定义main方法 ，在方法体中  
    创建Person类对象person,并给其name属性赋值为"jack"  
    调用person对象的eat方法  
    创建Tiger类对象tiger,并给其name属性赋值为"tom"  
    调用tiger对象的eat方法  
    程序分析：  
    \*/  
    public static void main(String[] args) {  
    Person person = new Person();  
    person.name = "jack";  
    person.eat();  
    Tiger tiger = new Tiger();  
    tiger.\_\_name\_\_\_ = "tom";  
    tiger.eat();  
    }  
   }  
   请按照以上需求补全代码，使得程序运行，控制台输出如下  
   jackeatfood  
   tomeatmeat

作答结果：✓

正解：

①

②

③

④

解析：

## JAVASE(12-1.构造方法)

试卷得分：100分

多选题

1.下列说法正确的是（ ）

 构造方法无法通过该类对象对象名. 去调用  
 构造方法可以通过该类对象对象名. 去调用  
 构造方法小写开头  
 构造方法大写开头

作答结果：✓

正解：AD

解析：

2.下列说法正确的是（ ）

 当一个类没有显示的声明构造函数,会默认添加一个隐式的满参构造函数  
 当一个类没有显示的声明构造函数,会默认添加一个隐式的无参构造函数  
 当显示声明了构造方法后，隐式构造函数会消失  
 当显示声明了构造方法后，隐式构造函数不会消失

作答结果：✓

正解：BC

解析：

3.请选择下列说法中正确的()

 一个类中构造方法只能定义一个  
 一个类中构造方法可以定义多个  
 构造方法用来解决为对象属性赋初值的问题  
 构造方法不可重载

作答结果：✓

正解：BC

解析：

4. 请选择下列说法中正确的()

 创建子类对象，一定会调用父类构造方法  
 创建子类对象，可以不调用父类的构造方法  
 子类构造方法中，父类的构造方法只能被调用一次，并且放在第一行  
 子类构造方法中，父类的构造方法可以被调用多次

作答结果：✓

正解：AC

解析：

5.请选择正确的一项()  
  public class SuperType {  
   public SuperType(){  
   System.out.print("super");  
   }  
  }  
  public class SubType extends SuperType{  
   public SubType(){  
   System.out.print("sub");  
   }  
  }  
  public class Main {  
    
   public static void main(String[] args) {  
   SubType st = new SubType();  
   }  
    
  }

 报错  
 控制台输出supersub  
 控制台输出subsuper  
 控制台输出sub

作答结果：✓

正解：B

解析：

6.请选择下列说法正确的()

 构造方法与类名方法一致  
 构造方法没有返回值  
 创建对象时一定需要调用构造方法  
 创建对象时，如果没有显示声明构造方法，则不会调用构造方法

作答结果：✓

正解：ABC

解析：

7.请选择正确的一项()  
  public class SuperType {  
   public SuperType(){  
   System.out.print("super");  
   }  
  }  
  public class SubType extends SuperType{  
   public SubType(){  
   System.out.print("sub");  
   super();  
   }  
  }  
  public class Main {  
    
   public static void main(String[] args) {  
   SubType st = new SubType();  
   }  
    
  }

 报错  
 控制台输出supersub  
 控制台输出subsuper  
 控制台输出subsuper

作答结果：✓

正解：A

解析：

8.请选择正确的一项()  
  public class SuperType {  
   public SuperType(String param){  
   System.out.print("super");  
   }  
  }  
  public class SubType extends SuperType{  
   public SubType(){  
   super();  
   System.out.print("sub");  
   }  
  }  
  public class Main {  
    
   public static void main(String[] args) {  
   SubType st = new SubType();  
   }  
    
  }

 报错  
 控制台输出supersub  
 控制台输出subsuper  
 控制台输出sub

作答结果：✓

正解：A

解析：

9.请选择正确的一项()  
  public class Animal {  
   public int age;  
   public Animal(int age){  
   this.age = age;  
   }  
   public Animal(){  
   }  
  }  
  public class Person extends Animal{  
   public Person(int age) {  
   super();  
   }  
  }  
  public static void main(String[] args) {  
   Person p = new Person(10);  
   System.out.println(p.age);  
  }

 报错  
 控制台输出0  
 控制台输出10  
 控制台输出空白字符串

作答结果：✓

正解：B

解析：

10.请选择正确的一项()  
   package test;  
   public class FatherClass {  
     public FatherClass() {  
             System.out.println(“FatherClass Create”);  
     }  
   }  
   package test;  
   import test.FatherClass;  
   public class ChildClass extends FatherClass {  
          public ChildClass() {  
             System.out.println(“ChildClass Create”);  
     }  
     public static void main(String[] args) {  
               FatherClass fc = new FatherClass();  
               ChildClass cc = new ChildClass();  
            }  
   }

 ChildClass Create ChildClass Create FatherClass Create  
 ChildClass Create FatherClass Create FatherClass Create  
 FatherClass Create FatherClass Create ChildClass Create  
 FatherClass Create ChildClass Create FatherClass Create

作答结果：✓

正解：C

解析：

## JAVASE(13-1.包的概念)

试卷得分：90分

多选题

1.下列关于正确的是（ ）

 包名应该小写  
 包名应该大写  
 包名应在类文件的第一行声明  
 包可以在类文件的任何一行声明

作答结果：✓

正解：AC

解析：

2.下列说法正确的是（ ）

 当前类在哪个包下面，应使用关键字 import定义  
 当前类在哪个包下面，应使用关键字 package定义  
 java编译器会自动导入java.lang包  
 java编译器会自动导入java.util包

作答结果：✓

正解：BC

解析：

3.如果使用到一个类，但是没有导入，则该类一定存在于()

 java.io  
 java.util  
 java.lang  
 java.awt

作答结果：✓

正解：C

解析：

4.请选择下列正确的定义()

 public class Person{ import test; }  
 import test; public class Person{ }  
 public class Person{ } import test;  
 import test public class Person{ }

作答结果：✓

正解：B

解析：

5.请选择定义正确的一项()

 package aaa; import bbb; public class Xxx{}  
 import bbb; package aaa; public class Xxx{}  
 import bbb; public class Xxx{} package aaa;  
 package aaa; public class Xxx{} import bbb;

作答结果：✓

正解：A

解析：

6.请选择下列说法正确的()

 处于相同包下面不同类之间相互使用，需要import  
 处于相同包下面不同类之间相互使用，不需要import  
 处于不同包下面不同类之间相互使用，需要import  
 处于不同包下面不同类之间相互使用，不需要import

作答结果：✓

正解：BC

解析：

7.请选择下列说法正确的

 File类在java.io包下面  
 FileInputStream类在java.io包下面  
 File类在java.util包下面  
 FileInputStream类在java.util包下面

作答结果：✓

正解：AB

解析：

8.请选择下列说法正确的 ()

 ArrayList类在java.io包下面  
 HashMap类在java.io包下面  
 ArrayList类在java.util包下面  
 HashMap类在java.util包下面

作答结果：✓

正解：CD

解析：

9.请选择下列说法正确的()

 java.net包下面的用于JAVA桌面程序开发  
 java.util.Date 包下面的类用于JAVA桌面程序开发  
 快速导包的快捷键是 ctrl + shift + o  
 快速导包的快捷键是 ctrl + shift + r

作答结果：✓

正解：C

解析：

10.请选择下列没有编译错误的选项()

 package aa; public class A { } package aa.bb; import aa.A; public class B { public A a = new A(); } package bb; import aa.bb.B; public class C { public B b = new B(); }  
 package aa; public class A { } package aa.bb; public class B { public A a = new A(); } package bb; import aa.bb.B; public class C { public B b = new B(); }  
 package aa; public class A { } package aa.bb; import aa.A; public class B { public A a = new A(); } package bb; public class C { public B b = new B(); }  
 package aa; public class A { } package aa.bb; public class B { public A a = new A(); }

作答结果：☓

正解：A

解析：

## JAVASE(14-1.访问修饰符与get、set方法)

试卷得分：70分

多选题

1.下列关于说法正确的是（ ）

 一般类可以用public 修饰  
 一般类可以用private 修饰  
 一般类可以用protected 修饰  
 当没有显示添加权限修饰符类的情况下，类默认使用public 修饰

作答结果：✓

正解：A

解析：

2.下列说法错误的是（ ）

 类里面不可以再定义类  
 类里面可以定义类  
 同一个java文件中，可以有多个公共类  
 同一个java文件中，只能有一个公共类

作答结果：✓

正解：AC

解析：

3.下列说法正确的是()

 public 修饰符=＞当前类可以访问，其他类也可访问/继承  
 public 修饰符=＞当前类可以访问，其他类不可访问/继承  
 private 修饰符=＞当前类可以访问，其他类也可访问/继承  
 private 修饰符=＞当前类可以访问，其他类不可访问/继承

作答结果：✓

正解：AD

解析：

4.下列说法正确的是()

 protected 修饰符=＞同包下的类可以访问/继承，其他类不可访问  
 protected修饰符=＞同包下的类不可访问/继承，其他类可以访问  
 protected 修饰符=＞同包下的类可以访问/继承，其他类不可继承  
 protected修饰符=＞同包下的类不可访问/继承，其他类可以继承

作答结果：☓

正解：A

解析：

5.下列说法正确的是()

 default 修饰符=＞同包下的类可以访问/继承，其他类不可访问  
 default修饰符=＞同包下的类不可访问/继承，其他类可以访问  
 default修饰符=＞同包下的类不可以访问/继承，其他类不可继承  
 default修饰符=＞同包下的类可访问/继承，其他类可以继承

作答结果：☓

正解：D

解析：

6.请选择下列setget方法定义正确的()

 public class Person { private int age; public int getAge() { return age; } public void setAge(int age) { this.age = age; } }  
 public class Person { public int age; public int getAge() { return age; } public void setAge(int age) { this.age = age; } }  
 public class Person { private int age; public int getAge() { return age; } public void setAge(String age) { this.age = age; } }  
 public class Person { private int age; public void getAge() { return age; } public int setAge(int age) { this.age = age; } }

作答结果：✓

正解：A

解析：

7.请选择下列说法正确的

 子类可以继承父类所有方法和属性  
 子类可以继承父类非private方法和非private属性  
 无论是否处于同包，子类都不可以继承父类protected方法和属性  
 无论是否处于同包，子类都不可以继承父类default方法和属性

作答结果：✓

正解：B

解析：

8.请选择下列定义最正确的一项 ()

 public class Person { private boolean right; public boolean setRight() { return right; } public void isRight(boolean right) { this.right = right; } }  
 public class Person { private boolean right; public boolean isRight() { return right; } public void setRight(boolean right) { this.right = right; } }  
 public class Person { public boolean right; public boolean getRight() { return right; } public void setRight(boolean right) { this.right = right; } }  
 public class Person { private boolean right; public boolean getRight (boolean right) { return right; } public void () setRight { this.right = right; } }

作答结果：✓

正解：B

解析：

9.请选择下列说法正确的()

 @Data注解可以自动生成该类私有属性的set get 方法  
 @Data注解可以自动导入类里边引用的所有外部类  
 使用@Data注解，应导入lombok包  
 使用@Data注解，应导入java.util包

作答结果：✓

正解：AC

解析：

10.请选择下列setget方法定义正确的()

 public class Person { private String name; private int age; public String getName() { return name; } public void setName(String name) { this.name = name; } public int getAge() { return age; } public void setAge(int age) { this.age = age; } }  
 public class Person { private String name; private int age; public String getName() { return name; } public void setAge(int age) { this.age = age; } }  
 public class Person { private String name; private int age; public String getName() { return name; } public void setAge(String name) { this.name = name; } public int getAge() { return age; } public void setName(int age) { this.age = age; } }  
 public class Person { private String name; private int age; public String getAge() { return name; } public void setName(String name) { this.name = name; } public int getName() { return age; } public void setAge(int age) { this.age = age; } }

作答结果：☓

正解：A

解析：

## JAVASE(15-1.抽象类)

试卷得分：70分

多选题

1.下列说法正确的是（ ）

 当方法内容不确定时,可以使用abstract关键字来修饰，并且没有方法体  
 当方法内容不确定时,可以使用abstract关键字来修饰，并且拥有方法体  
 用abstract修饰的方法，一定是抽象方法  
 用abstract修饰的方法，不一定是抽象方法

作答结果：✓

正解：AC

解析：

2.下列说法正确的是（ ）

 java中只有抽象类和抽象方法，没有抽象属性  
 java中有抽象类和抽象方法，也有抽象属性  
 java中有抽象类和抽象方法，也有抽象属性  
 有抽象方法的类一定是抽象类

作答结果：✓

正解：AD

解析：

3.下列抽象方法定义正确的是()

 public void method();  
 public void method(){}  
 public abstract void method(){}  
 public abstract void method();

作答结果：✓

正解：D

解析：

4.下列抽象类定义正确的是()

 public class abstract Person {}  
 public abstract class Person {}  
 public class Person {}  
 abstract public class Person {}

作答结果：☓

正解：B

解析：

5.下列说法正确的是()

 抽象类一定有抽象方法。  
 抽象类不一定拥有抽象方法  
 抽象类中不可以拥有普通方法  
 抽象类中可以拥有普通方法

作答结果：☓

正解：BD

解析：

6.下列说法正确的是()

 抽象类中可以拥有构造函数  
 抽象类中不可以拥有构造函数  
 抽象类可以被实例化  
 抽象类不可以被实例化

作答结果：✓

正解：AD

解析：

7.请选择下列说法正确的

 抽象类不可以被继承  
 抽象类可以被继承  
 抽象类可以拥有一般成员属性  
 抽象类不可以拥有一般成员属性

作答结果：☓

正解：BC

解析：

8.请选择下列抽象类定义正确是 ()

 public abstract class Animal { public int age; public abstract void walk(); public void eat(){} }  
 public abstract class Animal { public int age; public abstract void walk(); public void eat(); }  
 public abstract class Animal { public abstract int age; public abstract void walk(); public void eat(){} }  
 public abstract class Animal { public int age; public abstract void walk(){}; public void eat(){} }

作答结果：✓

正解：A

解析：

9.请选择下列中没有编译错误的()

 public abstract class Animal { public abstract void walk(); public void eat(){} } class Person extends Animal{ public void walk(); }  
 public abstract class Animal { public abstract void walk(); public void eat(){} } class Person extends Animal{ public void walk() {} }  
 public abstract class Animal { public abstract void walk(); public void eat(){} } abstract class Person extends Animal{}  
 public abstract class Animal { public abstract void walk(); public void eat(){} } abstract class Person extends Animal{ public void walk(){} }

作答结果：✓

正解：BCD

解析：

10.请选择下列正确的结果()  
   public abstract  class Animal {  
    public String name;  
    public abstract void walk();  
    public Animal(String name){  
    this.name = name;  
    }  
   }  
     
   public class Person extends Animal{  
    public Person(String name) {  
    super(name);  
    }  
    public  void walk(){  
    System.out.println("human walk");  
    }  
   }  
     
     
   public class Main {  
    public static void main(String[] args) {  
    Person person = new Person("jack");  
    person.walk();  
    }  
   }

 编译错误  
 运行错误  
 运行控制台输出human walk  
 不报错控制台也没有任何输出

作答结果：✓

正解：C

解析：

## JAVASE(16-1.接口)

试卷得分：90分

多选题

1.下列说法正确的是（ ）

 JAVA中使用abstract关键字定义接口  
 JAVA中使用interface关键字定义接口  
 JAVA中接口中只能定义抽象方法  
 JAVA中接口中可以定义一般方法

作答结果：☓

正解：BC

解析：

2.下列说法正确的是（ ）

 JAVA中接口中可以定义普通属性  
 JAVA中接口中可以定义静态属性  
 JAVA中接口中一定要定义至少一个方法，否则编译报错  
 JAVA中接口中可以不定义方法，编译不报错

作答结果：✓

正解：BD

解析：

3.下列关于接口定义正确的是()

 public interface IStudentService{}  
 interface public IStudentService{}  
 public class IStudentService{}  
 public IStudentService{}

作答结果：✓

正解：A

解析：

4.下列说法正确的是（ ）

 JAVA中接口中定义方法必须显式声明权限修饰符 public 或 protected  
 JAVA中 接口中定义方法必须显式声明权限修饰符 public  
 JAVA中接口中定义方法可以不显式声明权限修饰符,隐式指定为public  
 JAVA中接口中定义方法可以不显式声明权限修饰符,隐式指定为public或 protected

作答结果：✓

正解：C

解析：

5.下列说法正确的是()

 JAVA中接口中可以拥有构造方法  
 JAVA中接口不能被实例化  
 JAVA中，一个接口，不能继承多个接口。  
 JAVA中一个类，只能实现一个接口

作答结果：✓

正解：B

解析：

6.下列关于接口定义正确的是()

 public interface IStudentService { protected void method(); }  
 public interface IStudentService { public void method(){} }  
 public interface IStudentService { private void method(); }  
 public interface IStudentService { void method(); }

作答结果：✓

正解：D

解析：

7.请选择下列说法正确的

 JAVA中接口中的方法必须显示声明为abstract  
 JAVA中接口中的方法无需声明为abstract ，默认为abstract修饰  
 JAVA中接口支持多继承  
 JAVA中接一个类可实现多个接口

作答结果：✓

正解：BCD

解析：

8.请选择下列接口类定义正确的是 ()

 public interface IStudentService { String serviceMehotd1; void serviceMehotd2(); void serviceMehotd3(); }  
 public interface IStudentService { String serviceMehotd1; void serviceMehotd2(); void serviceMehotd3(); }  
 public interface IStudentService { String serviceMehotd1(); void serviceMehotd2(); void serviceMehotd3(){} }  
 public interface IStudentService { String serviceMehotd1(); void serviceMehotd2(); void serviceMehotd3(); }

作答结果：✓

正解：D

解析：

9. 以下为接口定义，请选择该接口正确的实现类()  
  public interface IStudentService {  
    String method1(String a);  
    int method2(int b);  
  } 

 public class IStudentServiceImpl implements IStudentService { public int method1(String a) { return null; } public String method2(int b) { return 0; } }  
 public class IStudentServiceImpl implements IStudentService { public String method1(String b) { return null; } public int method2(int a) { return 0; } }  
 public class IStudentServiceImpl implements IStudentService { public void method1(String a) {} public void method2(int b) {} }  
 public class IStudentServiceImpl implements IStudentService { public void method1() {} public void method2() {} }

作答结果：✓

正解：B

解析：

10.以下为接口定义，请选择该接口正确的实现类()  
   public interface IService1 {  
    void method1();  
   }  
   public interface IService2 extends IService1{  
    int method2();  
    String method3();  
   }  
   

 public class IService2Impl implements IService2 { public void method1() {} public void method2() {} public String method3() { return null; } }  
 public class IService1Impl implements IService2 { public int method2() { return 0; } public String method3() { return null; } }  
 public class IService1Impl implements IService2 { public void method1() {} public int method2() { return 0; } public String method3() { return null; } }  
 public class IStudentServiceImpl implements IStudentService { public String method2() { return 0; } public int method3() { return null; } }

作答结果：✓

正解：C

解析：

## JAVASE(17-1.值传递和引用传递)

试卷得分：90分

多选题

1.请选择下列说法正确的是（ ）

 值传递作为参数传递给方法后，不随方法体中值的变化而变化  
 值传递作为参数传递给方法后，随方法体中值的变化而变化  
 引用传递作为参数传递给方法后，不随方法体中值的变化而变化  
 引用传递作为参数传递给方法后，随方法体中值的变化而变化

作答结果：✓

正解：AD

解析：

2.请选择控制台输出结果（ ）  
  public static void main(String [] args){  
        int  a   =   10;  
  changeValue(a);  
        System.out.println(a);  
  }  
  public static void changeValue(int a ){  
        System.out.println(a);  
        a =  100;  
  }

 先打印10 再打印10  
 先打印100 再打印100  
 先打印10 再打印100  
 先打印100 再打印10

作答结果：✓

正解：A

解析：

3.请选择控制台输出结果（ ）  
  public static void main(String [] args){  
     String  str   =   ‘123’;  
  changeValue(str);  
          System.out.println(str);  
  }  
  public static void changeValue(String str ){  
            System.out.println(str);  
            str =  ‘456’;  
  }

 先打印123 再打印123  
 先打印123 再打印456  
 先打印456再打印456  
 先打印456 再打印123

作答结果：✓

正解：A

解析：

4.请选择控制台输出结果（ ）  
  public static void main(String [] args){  
     Integer  a   =   10;  
  changeValue(a);  
           System.out.println(a);  
  }  
  public static void changeValue(Integer  a ){  
          System.out.println(a);  
           a = 100;  
  }

 先打印10 再打印100  
 先打印100 再打印100  
 先打印10 再打印10  
 先打印100 再打印10

作答结果：✓

正解：C

解析：

5.请选择控制台输出结果（ ）  
  public class Run{  
       public static void main(String [] args){  
               Person person = new Person( 10);  
               changeValue(person );  
               System.out.println(person.age);  
  }  
  public static void changeValue(Person  person ){  
                    System.out.println(person.age);  
                    person.age = 100;  
  }  
  }  
  class Person{  
         public int age;  
         public Person(int age){  
                this.age = age;  
        }  
  }

 先打印100 再打印100  
 先打印10 再打印100  
 先打印10 再打印10  
 先打印100 再打印10

作答结果：✓

正解：B

解析：

6.请选择说法正确的一项（ ）

 值类型作为参数传递给方法后，不随方法体中值的变化而变化  
 值类型作为参数传递给方法后，随方法体中值的变化而变化  
 8个基本类型封装类作为参数传递给方法后，不随方法体中值的变化而变化  
 8个基本类型封装类座位参数传递给方法后，随方法体中值的变化而变化

作答结果：✓

正解：AC

解析：

7.请选择说法正确的一项（ ）

 String类型作为参数传递给方法后，不随方法体中值的变化而变化  
 String作为参数传递给方法后，随方法体中值的变化而变化  
 数组类型作为参数传递给方法后，不随方法体中值的变化而变化  
 数组类型作为参数传递给方法后，随方法体中值的变化而变化

作答结果：✓

正解：AD

解析：

8.请选择控制台输出结果（ ）  
  public static void main(String [] args){  
     int[]  arr   =   {10,100};  
  changeValue(arr);  
           System.out.println(arr[0]);  
  }  
  public static void changeValue(int[]  arr){  
          System.out.println(arr[0]);  
          arr[0] = 100;  
  }

 先打印10 再打印100  
 先打印100 再打印100  
 先打印10 再打印10  
 先打印100 再打印10

作答结果：✓

正解：A

解析：

9.请选择控制台输出结果（ ）  
  public static void main(String [] args){  
          String  str   =  new String(‘123’);;  
  changeValue(str);  
          System.out.println(str);  
  }  
  public static void changeValue(String str ){  
          System.out.println(str);  
          str =  ‘456’;  
  }

 先打印123 再打印123  
 先打印123 再打印456  
 先打印456再打印456  
 先打印456 再打印123

作答结果：✓

正解：A

解析：

10.请选择控制台输出结果（ ）  
   public class Run{  
        public static void main(String [] args){  
              Person person = new Person(10);  
              changeValue(person);  
              System.out.println(person.age);  
   }  
   public static void changeValue(Person  person ){  
                     System.out.println(person.age);  
                     person  = new Person();  
                     person.age = 100;  
   }  
   }  
   class Person{  
          public int age;  
          public Person(int age){  
                 this.age = age;  
         }  
         public Person(){  
         }  
   }

 先打印10 再打印100  
 先打印100 再打印100  
 先打印10 再打印10  
 先打印100 再打印10

作答结果：☓

正解：C

解析：

## JAVASE(18-1.final关键字)

试卷得分：100分

多选题

1.请选择下列说法正确的（ ）

 用final关键字修饰的属性/变量可以不赋初始值  
 用final关键字修饰的属性/变量必须赋初始值  
 用final关键字修饰的方法不可以被重写  
 用final关键字修饰的方法可以被重写

作答结果：✓

正解：BC

解析：

2.请选择正确的一项（ ）  
  public static final int  a  = 10;  
  public static void main(String [] args){  
       a   =   100;  
       System.out.println(a);  
  }

 打印10  
 打印100  
 编译报错  
 运行报错

作答结果：✓

正解：C

解析：

3.请选择下列说法正确的（ ）

 用final关键字修饰的类不能被继承  
 用final关键字修饰的类可以被继承  
 java.lang.String类可以被继承  
 用final关键字修饰的方法可以被继承

作答结果：✓

正解：AD

解析：

4.请选择下列说法正确的（ ）

 final关键字可以用来修饰一个类  
 final关键字不可以用来修饰一个方法  
 final关键字不可以修饰成员变量  
 final关键字可以修饰静态变量

作答结果：✓

正解：AD

解析：

5.请选择下列说法错误的是（ ）

 final关键字不可以局部变量  
 final关键字不可以修饰类  
 final关键字可以修饰成员变量  
 final关键字可以修饰静态变量

作答结果：✓

正解：AB

解析：

6.请选择正确的一项（ ）  
  package run;  
  public class Main {  
   public static void main(String [] args){  
   Person person = new Person();  
   person.age = 100;  
   System.out.println(person.age);  
   }  
  }  
  class Person{  
   public final int age =10;  
  }

 打印10  
 打印100  
 编译报错  
 运行报错

作答结果：✓

正解：C

解析：

7.请选择正确的一项（ ）  
  package run;  
  public class Main {  
   public static void main(String [] args){  
   Person person = new Person();  
   person.age = 100;  
   System.out.println(person.age);  
   }  
  }  
  final class Person{  
   public int age =10;  
  }

 打印10  
 打印100  
 编译报错  
 运行报错

作答结果：✓

正解：B

解析：

8.请选择正确的一项（ ）  
  package run;  
    
  public  class Main {  
   public static void main(String [] args){  
   final Person person = new Person();  
   person.age = 100;  
   System.out.println(person.age);  
   }  
  }  
  class Person{  
    public int age = 10;  
  }

 打印10  
 打印100  
 编译报错  
 运行报错

作答结果：✓

正解：B

解析：

9.请选择正确的一项（ ）  
  package run;  
    
  public  class Main {  
   public static void main(String [] args){  
   final Person person = new Person();  
   person.age = 100;  
   person = new Person();  
   System.out.println(person.age);  
   }  
  }  
  final class Person{  
    public int age = 10;  
  }

 打印10  
 打印100  
 编译报错  
 运行报错

作答结果：✓

正解：C

解析：

10.请选择正确的一项（ ）  
   package run;  
     
   public  class Main {  
    public static void main(String [] args){  
    Person person = new Person();  
    person = new Person();  
    System.out.println(person.dog.age);  
    }  
   }  
   class Person{  
     public  final Dog dog = new Dog(100) ;  
   }  
   final class Dog{  
    public Dog(int age){  
     
    }  
    public final int age = 10;   
   }

 打印10  
 打印100  
 编译报错  
 运行报错

作答结果：✓

正解：A

解析：

## JAVASE(19-1.static关键字(静态关键字))

试卷得分：93分

多选题

1.请选择下列说法正确的是（ ）

 .static关键字可以修饰属性  
 static关键字可以修饰方法  
 static关键可以修饰类  
 static关键可以修饰接口

作答结果：✓

正解：AB

解析：

2.请选择下列说法正确的是（ ）

 static方法的方法体中，只能调用static方法和static属性  
 static方法的方法体中，可以调用成员方法和成员属性  
 static方法中，可以使用this关键字  
 static方法中，可以使用super关键字

作答结果：✓

正解：A

解析：

3.请选择下列说法正确的（ ）

 用final关键字修饰的类不能被继承  
 用final关键字修饰的类可以被继承  
 java.lang.String类可以被继承  
 用final关键字修饰的方法可以被继承

作答结果：☓

正解：AD

解析：

4.如果一个类的成员变量只能在所在类中使用， 则该成员变量必须使用 的修饰是()

 public  
 protected  
 private  
 static

作答结果：✓

正解：C

解析：

5.关于抽象方法的说法正确的是

 可以有方法体  
 可以出现在非抽象类中  
 是没有方法体的方法  
 抽象类中的方法都是抽象方法

作答结果：✓

正解：C

解析：

6. Java application中的主类需包含 main 方法，以下哪项是 main 方法的 正确形参？（ ）

 String args  
 String args[]  
 Char arg  
 StringBuffer args[]

作答结果：✓

正解：B

解析：

7. return 语句： （ ）

 只能让方法返回基本数据类型  
 方法不一定有返回值  
 多句 return可以返回多个值  
 不能用来返回引用数据类型

作答结果：✓

正解：B

解析：

8.为 AB类的一个无形式参数无返回值的方法 method 书写方法头，使 得使用类名 AB作为前缀就可以调用它，该方法头的形式为 ()

 static void method( )  
 public void method( )  
 final void method( )  
 abstract void method( )

作答结果：✓

正解：A

解析：

9.下面程序段执行后 b 的值是 ( )  
  Integer integ =new Integer(9)；  
  boolean b = integ instanceof Object;

 9.0  
 true  
 1.0  
 0

作答结果：✓

正解：B

解析：

10.关于接口哪个正确？ （ ）

 实现一个接口必须实现接口的所有方法  
 一个类只能实现一个接口  
 接口间不能有继承关系  
 接口和抽象类是同一回事

作答结果：✓

正解：A

解析：

11.注释的基本原则不包括： （ ）

 注释应该增加代码的清晰度  
 注释要简洁  
 在写代码之前写注释  
 尽量给每一条语句加注释

作答结果：✓

正解：D

解析：

12.Java中()

 一个子类可以有多个父类，一个父类也可以有多个子类  
 一个子类可以有多个父类，但一个父类只可以有一个子类  
 一个子类可以有一个父类，但一个父类可以有多个子类  
 上述说法都不对

作答结果：✓

正解：C

解析：

13.关于以下程序段，正确的说法是（    ）C  
    1．  String  s1=”abc”+”def”;   
   2．  String  s2=new  String（s1）；  
    3．  if（s1= =s2）    
   4．     System.out.println(“= = succeeded”);    
   5．  if (s1.equals(s2))    
   6．     System.out.println(“.equals()  succeeded”); 

 行4与行6都将执行  
 行4执行，行6不执行  
 行6执行，行4不执行  
 行4、行6都不执行

作答结果：✓

正解：C

解析：

14.public class Test {  
    static void main(String[] args) {  
     System.out.print("aaa");  
    }  
   }  
   下面说法正确的是?

 程序无法通过编译，因为main方法声明方式不对  
 程序可以通过编译，但运行程序行，会提示": 在类Test 中找不到 main 方法, 请将 main 方法定义为:...."的错误  
 程序正常运行，并输出aaa  
 程序可以编译，但无法运行，因为该文件没有public类

作答结果：✓

正解：B

解析：

15.class HasStatic{  
    private static int x=100;  
    public static void main(String args[ ]){  
     HasStatic hs1=new HasStatic( );  
     hs1.x++;  
     HasStatic hs2=new HasStatic( );  
     hs2.x++;  
     hs1=new HasStatic( );  
     hs1.x++;  
     HasStatic.x--;  
     System.out.println("x="+x);  
    }  
   }

 程序通过编译，输出结果为：x=99  
 程序通过编译，输出结果为：x=101  
 程序通过编译，输出结果为：x=103  
 程序通过编译，输出结果为：x=102

作答结果：✓

正解：D

解析：

## JAVASE(20-1.集合List和Set)

试卷得分：85分

多选题

1.请选择下列说法正确的是（ ）

 List有序可以重复  
 List有序不可重复  
 Set无序可重复  
 Set无序不可重复

作答结果：✓

正解：AD

解析：

2.请选择下列说法正确的是（ ）

 List长度可变  
 Array长度可变  
 List中可以存放基本数据类型  
 List中不可以存放引用数据类型

作答结果：✓

正解：A

解析：

3.请选择下列中没有编译错误的是（ ）

 List list = new List();  
 ArrayList list = new List();  
 List list = new ArrayList ();  
 ArrayList list = new ArrayList ();

作答结果：✓

正解：CD

解析：

4.请选择下列中没有编译错误的是（ ）

 Set set = new HashSet();  
 HashSet set = new Set ();  
 Set set = new Set ();  
 HashSet set = new HashSet();

作答结果：✓

正解：AD

解析：

5.下列关于抽象方法说法正确的是 ( )

 不能出现在正常类当中  
 可以出现在非抽象类中  
 是有方法体的方法  
 抽象类中的方法都是抽象方法

作答结果：✓

正解：A

解析：

6.欲构造ArrayList类得一个实例，此类继承了List接口，下列正确的是

 .ArrayList myList = new Object();  
 List myList = new ArrayList();  
 ArraylList myList = new List();  
 List myList = new List();

作答结果：✓

正解：B

解析：

7.实现下列哪个接口，可以启用比较功能(     )

 Runnable接口  
 Iterator接口  
 Serializable接口  
 Comparator接口

作答结果：✓

正解：D

解析：

8.下面代码运行的结果是(     )  
  ArrayList＜String＞ al = newArrayList＜String＞();  
  al.add(true);  
  al.add(123);  
  al.add(“abc”);  
  System.out.println(al);

 编译失败  
 [true,123]  
 [true,123,abc];  
 [abc];

作答结果：✓

正解：A

解析：

9.下面类或者接口中，不属于集合体系的是(      )

 java.util.Collections  
 java.util.Map  
 java.util.Vector  
 java.util.Hashtable

作答结果：☓

正解：B

解析：

10.题示代码的功能为：对于一个存放Person对象的ArrayList进行循  
   环遍历。并输出每个Person对象的idCard和userName。 (  )C  
   public class Person{  
                     privateLong idCard;  
                     pirvateString userName;  
                     //以下是getter和setter方法  
                     //省略  
   }  
   List＜Person＞ list=newArrayList＜＞();   
   Person p1=new Person();  
   p1.setIdCard(new Long(1001));  
   p1.setUserName(“terry”); Person p2=new Person();  
   p2.setIdCard(new Long(1002));  
   p2.setUserName(“tom”);  
   list.add(p1);  
   list.add(p2);  
   for( 位置① ){  
   System.out.println(person.getIdCard()+”:”+person.getUserName());  
   }  
   那么位置①处的代码为 

 List list:person  
 List person:Person  
 Person person:List  
 Person person:list

作答结果：✓

正解：D

解析：

11.将集合转成数组的方法是(     )

 asList()  
 toCharArray()  
 toArray()  
 copy()

作答结果：✓

正解：C

解析：

12.在JAVA中ArrayList类实现了可变大小的数组，便于遍历元素和随机访问元素，已知获得了ArrayList类的对象bookTypeList，则下列语句中能够实现判断列表中是否存在字符串“小说”的是（  ）

 bookTypeList.add("小说");  
 bookTypeList.get("小说");  
 bookTypeList.contains("小说");  
 bookTypeList.remove("小说");

作答结果：☓

正解：CD

解析：

13.下面关于Collection和 Collections的区别正确的是(    )

 Collection是集合顶层接口  
 Collection是针对Collections集合操作的工具类  
 List、Set、Map都继承自Collection接口  
 Collections是针对Collection集合操作的工具类

作答结果：✓

正解：AD

解析：

14.关于迭代器说法错误的是()

 迭代器是取出集合元素的方式  
 迭代器的hasNext()方法返回值是布尔类型  
 List集合有特有迭代器  
 next()方法将返回集合中的上一个元素.

作答结果：✓

正解：CD

解析：

## JAVASE(21-1.集合List和Set进阶)

试卷得分：71分

多选题

1.请选择下列说法正确的是（ ）

 List有序可以重复  
 List有序不可重复  
 Set无序可重复  
 Set无序不可重复

作答结果：✓

正解：AD

解析：

2.请选择下列说法正确的是（ ）

 List长度可变  
 Array长度  
 List中可以存放基本数据类型  
 List中只能存放引用数据类型

作答结果：☓

正解：A

解析：

3.请选择下列中没有编译错误的是（ ）

 List list = new List();  
 ArrayList list = new List();  
 List list = new ArrayList ();  
 ArrayList list = new ArrayList ();

作答结果：✓

正解：CD

解析：

4.请选择下列中没有编译错误的是（ ）

 Set set = new HashSet();  
 HashSet set = new Set ();  
 Set set = new Set ();  
 HashSet set = new HashSet();

作答结果：✓

正解：AD

解析：

5.下列说法错误的是 ( )

 List， Set， Map都继承自Collection接口  
 抽象方法不能出现在非抽象类中  
 抽象方法是没有方法体的方法  
 抽象方法是没有方法体的方法

作答结果：✓

正解：A

解析：

6.下列正确的是()

 ArrayList myList = new Object();  
 List myList = new ArrayList();  
 ArraylList myList = new List();  
 List myList = new List();

作答结果：✓

正解：B

解析：

7.下列说法正确的是 ( )

 List内部数据不可以重复  
 List内部数据可以重复  
 List可以添加相同地址的对象  
 List不可以添加相同地址的对象

作答结果：✓

正解：BC

解析：

8.下列说法错误的是 ( )

 Set通过equals和hashcode来判断内部数据是否重复  
 Integer和String已经重写了equals方法  
 我们通常不需要重写Set泛型类的equals方法  
 我们通常需要重写Set泛型类的equals方法

作答结果：☓

正解：C

解析：

9.下列说法错误的是 ( )

 Set通过equals和hashcode来判断内部数据是否重复  
 Integer和String已经重写了equals方法  
 我们通常不需要重写Set泛型类的equals方法  
 我们通常需要重写Set泛型类的equals方法

作答结果：☓

正解：C

解析：

10.下列说法正确的是 ( )

 集合常用方法中的get(index)方法用于添加某个元素  
 集合常用方法中的add(index)方法用于获取某个元素  
 集合常用方法中的isEmpty()方法用于删除元素  
 集合常用方法中的toArray()方法用于转换为数组

作答结果：✓

正解：D

解析：

11.下列说法正确的是 ( )

 集合常用方法中的get(index)方法用于添加某个元素  
 集合常用方法中的add(index)方法用于获取某个元素  
 集合常用方法中的isEmpty()方法用于删除元素  
 集合常用方法中的toArray()方法用于转换为数组

作答结果：✓

正解：D

解析：

12.下列说法错误的是 ( )

 集合常用方法中的removeAll(Collection e)用于删除两个集合的交集  
 集合常用方法中的contains(Object o)方法用于判断元素是否存在  
 集合常用方法中的toArray()方法用于已经创建实例，判断是否存在数据  
 集合常用方法中的clear()方法用于清除所有元素

作答结果：☓

正解：C

解析：

13.package run;  
     
   import java.util.ArrayList;  
   import java.util.List;  
     
   public class Main {  
    public static void main(String[] args) {  
    List＜Person＞ list = new ArrayList＜\_\_Person\_\_\_＞();  
    Person p1 = new Person(1,"张三");  
    Person \_\_p2\_\_\_ = new Person(2,"李四");  
    list.add(p1);  
    list.add(p2);  
    p1.age = 3;  
    System.out.println(list.get(0).age);  
     
    }  
   }  
   class Person{  
    public int age;  
    public String name;  
    public Person(int age, String name) {  
    super();  
    this.age = age;  
    this.name = name;  
    }  
   }  
     
   请补全代码，使得程序可以正常运行  
   并且程序运行控制台会输出\_\_3\_\_\_

作答结果：✓

正解：

①

②

③

解析：

14.package run;  
     
   import java.util.HashSet;  
   import java.util.Set;  
     
   public class Main {  
    public static void main(String[] args) {  
    Set＜Person＞ \_\_set\_\_\_ = new HashSet＜Person＞\_\_()\_\_\_;  
    Person p1 = new Person(1,"张三");  
    set.add(p1);  
    set.add(p1);  
    System.out.println(set.size());  
    }  
   }  
   class Person{  
    public int age;  
    public String name;  
    public Person(int age, String name) {  
    super();  
    \_\_this\_\_\_.age = age;  
    this.name = name;  
    }  
   }  
   请补全代码，使得程序可以正常运行  
   并且程序运行控制台会输出\_\_1\_\_\_

作答结果：✓

正解：

①

②

③

④

解析：

## JAVASE(22-1.HashMap讲解)

试卷得分：57分

多选题

1.请选择下列说法错误的是（ ）

 HashMap不可以将null作为key  
 HashMap可以将null作为key  
 LinkedHashMap 有序  
 LinkedHashMap 无序

作答结果：✓

正解：AD

解析：

2.请选择下列正确的（ ）

 HashTable 线程安全  
 HashTable 线程不安全  
 HashMap map = new Map();  
 Map map = new HashMap ();

作答结果：✓

正解：AD

解析：

3.请选择下列没有编译错误的是（ ）

 Map map = new HashMap＜String,String＞();  
 Map＜String,String＞ map = new HashMap();  
 Map＜String,String＞ map = new HashMap＜String,String＞ ();  
 HashMap ＜String,String＞ map = new Map＜String,String＞ ();

作答结果：☓

正解：BC

解析：

4.4. 请选择下列说法错误的是（ ）

 HashMap类的put方法只有一个参数  
 HashMap类的get方法无参  
 HashMap类的clear方法无参  
 HashMap类的remove方法无参

作答结果：✓

正解：ABD

解析：

5.下列说法正确的是 ( )

 HashMap的 key不能重复，value不能重复  
 HashMap的key不能重复，value可以重复  
 HashMap的key可以重复，value不能重复  
 HashMap的key可以重复，value可以重复

作答结果：✓

正解：B

解析：

6.package run;  
    
  import java.util.ArrayList;  
  import java.util.HashMap;  
  import java.util.List;  
  import java.util.Map;  
    
  public class Main {  
   public static void main(String[] args) {  
   //创建3个Student    
   Student stu1 = new Student(1,"张三",50);  
   Student stu2 = new Student(1,"李四",60);  
   Student stu3 = new Student(2,"王五",70);  
   List＜Student＞ dataSource = new ArrayList＜＞();  
   dataSource.add(stu1);  
   dataSource.add(stu2);  
   dataSource.add(stu3);  
   //按照年级分组后得到的map  
   Map＜Integer,List＜Student＞＞ map = new HashMap＜Integer,List＜Student＞＞();  
   //遍历数据源，按照年级分组  
   for(Student stu:dataSource){  
   if(map.containsKey(\_\_grade\_\_\_)){//第N次出现该年级  
   map.get(stu.grade).add(stu);  
   }else{//第一次出现该年级  
   List＜Student＞ \_\_newList\_\_\_ = new ArrayList＜＞();  
   newList.add(stu);  
   map.put(stu.grade, \_\_newList\_\_\_);  
   }  
   }  
   }  
  }  
  class Student{  
   //年级  
   public int grade;  
   //姓名  
   public String name;  
   //分数  
   public int score;  
    
   public Student(int grade, String name, int score) {  
   this.grade = grade;  
   this.name = name;  
   this.score = score;  
   }  
    
  }

作答结果：☓

正解：

①

②

③

解析：

7.package test;  
    
  import java.util.ArrayList;  
  import java.util.HashMap;  
  import java.util.List;  
  import java.util.Map;  
    
  public class HashMapTest {  
   public static void main(String[] args) {  
   //创建3个Student    
   Student stu1 = new Student("1","1","张三",50);  
   Student stu2 = new Student("2","1","李四",60);  
   Student stu3 = new Student("1","2","王五",70);  
   List＜Student＞ dataSource = new ArrayList＜＞();  
   dataSource.add(stu1);  
   dataSource.add(stu2);  
   dataSource.add(stu3);  
   //请按照(年级和班级)进行分组  
   //按照年级分组得到的变量(map)，其中key-＞grade,value-＞Map＜String,List＜Student＞(每个年级的数据容器)  
   //    value的Map＜String,List＜Student＞中， key-＞classNo，value-＞List＜Student＞(每个班级的数据容器)  
   Map＜String,Map＜String,List＜Student＞＞＞ map = new HashMap＜String,Map＜String,List＜Student＞＞＞();  
   //遍历数据源，按照年级分组  
   for(Student \_\_stu\_\_\_:dataSource){  
   if(map.containsKey(stu.grade)){//第N次出现该年级  
   if(map.get(stu.grade).containsKey(stu.classNo)){//第N次出现该班级  
   //添加到该班级分组  
   map.get(stu.grade).get(stu.classNo).add(stu);  
   }else{//第一次出现该班级  
   //新建班级班级分组  
   List＜Student＞ \_\_newList\_\_\_ = new ArrayList＜Student＞();  
   newList.add(stu);  
   map.get(stu.grade).put(stu.classNo, newList);  
   }  
    
   }else{//第一次出现该年级  
   Map＜String,\_\_List\_\_\_＞ newMap = new HashMap＜String,List＜Student＞＞();  
   map.put(stu.grade, newMap);  
   //年级第一次出现代表，班级也是第一次出现  
   List＜Student＞ newList = new ArrayList＜Student＞();  
   newList.add(stu);  
   newMap.put(stu.classNo, newList);  
   }  
   }  
   }  
    
  }  
  class Student{  
   //年级  
   public String grade;  
   //班级  
   public String classNo;  
   //姓名  
   public String name;  
   //分数  
   public int score;  
    
   public Student(String grade, String classNo, String name, int score) {  
   super();  
   this.grade = grade;  
   this.classNo = classNo;  
   this.name = name;  
   this.score = score;  
   }  
    
  }

作答结果：☓

正解：

①

②

③

解析：

## JAVASE(23-1.异常处理)

试卷得分：100分

多选题

1.请选择下列说法错误的是（ ）

 异常处理，是为了保证程序遇到异常时也能继续运行下去  
 异常处理，是为了增加程序的健壮性  
 异常和错误一样，程序员无法处理  
 异常分为运行时异常和编译时异常

作答结果：✓

正解：C

解析：

2.请选择下列说法正确的是（ ）

 运行时异常不需要提前处理，但为了健壮性最好添加处理  
 运行时异常隐式向上抛，把解决权交给调用端  
 运行时异常必须要手动添加throws声明  
 运行时异常无法通过手动抛出

作答结果：✓

正解：AB

解析：

3.请选择下列说法正确的是（ ）

 空指针异常属于运行时异常  
 类转换异常属于编译时异常  
 数组越界异常属于运行时异常  
 算法运算异常属于编译时异常

作答结果：✓

正解：AC

解析：

4.请选择下列说法正确的是（ ）

 catch语句可以单独存在  
 try和finally可以同时存在  
 finally语句可以单独存在  
 try语句，catch语句，finally语句必须三者同时存在

作答结果：✓

正解：B

解析：

5.程序运行控制台输出( )  
  public class Main {  
   public static void main(String[] args) {  
   int a = 1;  
   try{  
   System.out.println(1/0);  
   a += 1;  
   }catch(NullPointerException e){  
   a += 2;  
   }catch(ArithmeticException e){  
   a += 3;  
   }catch(Exception e){  
   a += 4;  
   }finally{  
   a += 5;  
   }  
   System.out.println(a);  
   }  
  }

 7  
 9  
 11  
 13

作答结果：✓

正解：B

解析：