

太戈编程模拟练习题

一、单项选择题（共 15 题，每题 2 分，共计 30 分，每题有且仅有一个正确选项）

1. 数据结构中，“先进后出”是（ ）结构的特征。

- A. 队列 B. 栈
C. 线性表 D. 树

2. 下面不同进制的数中最小的是（ ）。

- A. (11101101.01)₂ B. (323.87)₁₀
C. (371.56)₈ D. (2AF.4B)₁₆

3. 一个 8 位整型变量占用（ ）个字节

- A. 32 B. 8 C. 4 D. 1

4. 位水仙花数是指每个位上的数字的 3 次幂之和等于它本身。(例如： $1^3+5^3+3^3=153$)
找所有三位水仙花数程序如下，它的算法属于（ ）

```
for (i=1; i=9; i++);  
    for(j=0; j=9; j++);  
        for(k=0; k=9; k++);  
            if(i*i*i+j*j*j+k*k*k=i*100+j*10+k) cnt++;
```

- A. 动态规划 B. 递归 C. 枚举 D. 贪心

5. 下列四项中，合法的 IP 地址是（ ）

- A. 190 . 220 . 5 B. 206 . 53 . 78
C. 206 . 53 . 312 . 78 D. 123.43.82.220

6. 网址 <http://www.tinghua.edu.cn> 中的“cn”代表的是（ ）

- A. 域名 B. 服务器名 C. IP 地址 D. 国家

7. 把 7 个同样的球放入 4 个不同的袋子里，不允许有袋子空着不放，问一共有多少种不同的分法（ ）

- A. 22 B. 24 C. 18 D. 20

8. 如果约定二叉树根结点是第一层，那么以下哪项不能成为完全二叉树（ ）

- A. 2 层共 2 个结点 B. 4 层共 9 个结点
C. 5 层共 11 个结点 D. 6 层共 40 个结点

9. 12 以内的正整数（包含 12）互质的数共有（ ）对？(a,b)和(b,a)算同一对。

- A. 43 B. 44 C. 45 D. 46

10. 264 和 165 的最大公约数是 ()
A. 27 B. 31 C. 29 D. 33
11. 某班有 30 个同学报名参加 100、400、800m 3 个运动项目比赛。已知有 6 人获 100m 参赛资格, 8 人获 400m 参赛资格, 18 人获 800m 参赛资格。且其中有 1 人获全部 3 项参赛资格, 则至少有 () 人没有获任何项目参赛资格。
A. 0 B. 1 C. 2 D. 3
12. 假设今天是星期六, 再过 10^{2020} 天后的那一天是星期 ()
A. 一 B. 二 C. 三 D. 四
13. 下面有关计算机的特点叙述, 不正确的是 ()
A. 运算速度快
B. 有记忆和逻辑判断能力
C. 具有自动执行程序的能力
D. 至今没有任何人能给出如何求解方法的难题, 计算机也都能求出解来
14. 假设一棵二叉树的中序遍历序列为 DGBAECHF, 后序遍历序列为 GDBEHFCA, 则其前序遍历序列为 ()。
A. ABCDFGHE B. ABDGCEFH
C. ACBGDHEF D. ACEFHBGD
15. Facebook 是美国的一个社交网络服务网站, 它的创始人是()
A. 比尔盖茨
B. 史蒂夫乔布斯
C. 马克扎克伯格
D. 马特达蒙

二、阅读程序 (程序输入不超过数组或字符串定义的范围: 判断题正确填√, 错误填×; 除特殊说明外, 判断题 1.5 分, 选择题 3 分, 共计 40 分)

1.

1	#include<iostream>
2	using namespace std;
3	int main(){
4	int x,y;
5	cin>>x>>y;
6	int i=2,p=1;
7	do{
8	while(x%i==0&& y%i==0){
9	p*=i;
10	x/=i;
11	y/=i;

```

12     }
13     i++;
14     }while(x>=i&&y>=i);
15     cout<<p*x*y<<endl;
16     return 0;
17 }

```

假设输入的 x 和 y 都是正整数，完成下面的判断题和单选题

判断题：

1. 输入两个数字 x 必须大于等于 y ，否则则程序无输出。()
2. 若将第 6 行的 " $i=2$ " 改为 " $i=1$ "，程序输出变为原来的 $1/2$ 。()
3. 若将第 8 行的 " $x\%i==0\&\&y\%i==0$ " 改为 " $!(x\%i\&\&y\%i)$ "，程序运行结果不会改变。()
4. 若输入的数据互质，那么输出的数字为两者之积。()

选择题：

5. 若输入的数据为 88 32，那么输出为 ()。

A. 8 B. 352 C. 4 D. 32

6. 若输出结果是 280，则 () 不可能是输入数据

A. 140 2 B. 56 35 C. 40 7 D. 140 56

2.

```

1  #include<iostream>
2  using namespace std;
3  int a[101]={};
4  int main(){
5      int n,m; cin>>n>>m;
6      int p=1;
7      do{
8          a[p]++;
9          if(a[p]>n){
10             a[p]=0;
11             p--;
12         }
13         else{
14             p++;
15             a[p]=a[p-1];
16         }
17         if(p>m){
18             for(int i=1;i<=m;i++) cout<<a[i];
19             cout<<" ";
20             p--;

```

21	}
22	}while(p>=1);
23	return 0;
24	}

假设输入的 x 和 y 都是正整数，完成下面的判断题和单选题

判断题：

1. 输入两个数字 n 必须大于等于 m ，否则则程序无输出。()
2. 若将第 3 行和第 4 行置换，程序输出结果保持出不变。()
3. 若输入的数据为 9 5，则变量 p 在程序运行中不超过 5。()
4. 若输入的数据为 9 5，则输出所有数字串长度均不超过 5。()

选择题：

5. 若输入的数据为 5 3，则输出数字串的数量为()
 A. 5 B. 8 C. 10 D. 12

6. 若输入的数据为 4 2，那么输出为 ()。

A. 12 13 14 23 24 34 B. 12 13 14
 C. 23 24 34 D. 123 124 134 234

3.

1	#include<iostream>
2	using namespace std;
3	int b[6];
4	int main(){
5	for(int i=1;i<=5;i++) b[i]=i;
6	int s=1;
7	int j=1;
8	while(j>0){
9	j=5;
10	while(j>0&&b[j]==10+j-5) j--;
11	if(j>0){
12	s++;b[j]++;
13	for(int i=j+1;i<=5;i++)
14	b[i]=b[j]+i-j;
15	}
16	}
17	cout<<s<<endl;
18	return 0;
19	}

判断题：

1. 若将第 13 行的 "int i=j+1;i<=5;i++" 改为 "int i=j;i<=5;i++", 程序运行结果不会改变。()

2. 程序执行过程中, 数组 b 一直保持从小到大排列 (相邻可以相等)。()

选择题:

3. 程序执行完毕时, 下面哪个说法是正确的()。

A. b[1]=2 B. b[2]=10 C. b[3]=8 D. b[5]=5

4. 若将第 12 行的 "s++" 改为 "s+=2", 程序运行结果变为原来的 () 倍

A. 1 B. 2 C. 10 D. 4

5. 若将第 5 行的 "b[i]=i;" 改为 " () " 程序运行结果最小

A. b[i]=1; B. b[i]=i-1; C. b[i]=i+1; D. b[i]=i*2;

6. (4 分) 输出的值为 ()

A. 210 B. 252 C. 126 D. 462

三、完善程序 (单选题, 每小题 3 分, 共计 30 分)

1. (金蝉素数) 某古寺的一块石碑上依稀刻有一些三位或四位的神秘自然数。专家研究发现: 这些数字是素数, 且从低位去掉一位, 或两位,后都还是素数, 从高位去掉一位, 或两位,后也都仍为素数, 更奇妙的是, 同时去掉它的最高位与最低位数字后还是素数。因此, 人们把这些神秘的素数成为金蝉素数, 喻意金蝉脱壳之后仍然为美丽的金蝉。试求出石碑上的金蝉素数。

```

1  #include<iostream>
2  #include<cmath>
3  using namespace std;
4  int main(){
5      a[1]=2,a[2]=3,a[3]=5,a[4]=7;
6      int u=4;
7      for(int i=11;i<10000;i+=2){
8          int x=3;
9          while(x*x<=i&&____(1)____) x++;
10         if(x*x<=i)continue;
11         if(a[u]<1000)a[++u]=i//i 是素数
12         if(i<=100)continue;//三位及以上才行
13         int L=log(i)/log(10)+1;//i 的位数
14         int t=1,s=0;
15         for(int j=1;j<L;j++){
16             t*=10;
17             int hi=____(2)____;
18             int lo=____(3)____;
19             int v=1,mid=10000;

```

```

20         if(j==L-1)mid=lo/10;//去首去尾的中间部分
21         while(a[v]<=hi||a[v]<=lo){
22             if(a[v]==hi) s++;
23             if(a[v]==lo) s++;
24             if(a[v]==mid) s++;
25             ____ (4) ____;
26         }
27     }
28     if(____ (5) ____ ) cout<<i<<endl;
29 }
30 return 0;
312 }

```

1. (1) 处应填 ()。

A. $i\%x>0$ B. $i\%x<0$ C. $i\%x\geq 0$ D. $i\%x==0$

2. (2) 处应填 ()。

A. $i\%10$ B. $i/=10$ C. $i\%t$ D. i/t

3. (3) 处应填 ()。

A. $i\%10$ B. $i\%t$ C. i/t D. $i/=10$

4. (4) 处应填 ()。

A. $v=mid$ B. $v+=s$ C. $v++$ D. $s++$

5. (5) 处应填 ()。

A. $s=L$ B. $s=3||s=4$ C. $s=2*L-1$ D. $s=2*L$

2. (修理栅栏) 农夫约翰为了修理栅栏, 要将一块很长的木板切成 N 块。准备切成的木板长度为 L_1, L_2, \dots, L_N , 未切割前木板的长度恰好为切割后木板长度的总和。每次切断木板时, 需要的开销为这块木板的长度。例如长度为 21 的木板要切成长度为 5、8、8 的三块木板。长 21 的木板切成长度为 13 和 8 的板时, 开销为 21。再将长度为 13 的板切成长度为 5 和 8 的板时, 开销是 13。于是合计开销是 34。请求出按照目标要求将木板切割完最小的开销是多少?

试补全程序。

```

1  #include<iostream>
2  using namespace std;
3  typedef long long ll;
4  const int N=10009;
5  int n,L[N];
6  void solve(){
7      ll ans=0;
8      while(n>1){
9          int mii1=0,mii2=1;

```

```

10         if(L[mii1]>L[mii2]) swap(mii1,mii2);
11         for(int i=2;i<n;i++){
12             if(L[i]<L[mii1]){
13                 mii2=mii1;
14                 ____ (1) ____;
15             }
16             else if(____ (2) ____ ) mii2=i;
17         }
18
19         int t=____ (3) ____;
20         ans+=t;
21         if(mii1==n-1) swap(mii1,mii2);
22         L[mii1]=____ (4) ____;
23         L[mii2]=L[n-1];
24         ____ (5) ____;
25     }
26     cout<<ans<<endl;
27 }
28 int main(){
29     cin>>n;
30     for(int i=0;i<n;i++){
31         cin>>L[i];
32     }
33     solve();
34     return 0;
35 }

```

1. (1) 处应填 ()。

A. mii2=i B. mii1=i C. L[mii1]=L[i] D. L[mii2]=L[mii1]
2. (2) 处应填 ()。

A. L[i]>L[mii1] B. L[i]==L[mii1]
C. L[i]<L[mii2] D. L[i]>L[mii2]
3. (3) 处应填 ()。

A. L[mii1] B. L[mii1]+L[mii2] C. mii1+mii2 D. L[mii2]
4. (4) 处应填 ()。

A. t B. L[n-2] C. L[mii1] D. L[mii2]
5. (5) 处应填 ()。

A. n+=2 B. n-=2 C. n++ D. n--