

普及组模拟题第二套试题及答案

1. 以下不是属于国家顶级域名的是 (C)。

A. .au B. .cn C. .com D. .jp

【解析】.com 通用国际顶级域名, 国家顶级域名一般都是缩写, .au 是澳大利亚, .cn 是中国, .jp 是日本。

2. 2 个 10 进制数 1111 和 1010 的异或运算结果的 10 进制表示是 (C)。

A. 1958 B. 7 C. 1957 D. 9

【解析】参与运算的两个值, 如果两个相应 bit 位相同, 则结果为 0, 否则为 1。 $0^0 = 0$, $1^0 = 1$, $0^1 = 1$, $1^1 = 0$

1111 和 1010 是十进制数, 先转二进制 10001010111 和 1111110010, 异或运算结果 11110100101, 转二进制 1957

3. 8 位二进制数中去掉符号位, 最大能表示多少字符 (B)。

A. 127 B. 128 C. 255 D. 256

【解析】去掉符号位, 7 个 1, 为 127, 再算上 0 一共可以表示 128 个字符。

4. 在写递归函数时, 哪些定义一般不写在递归函数中? (D)

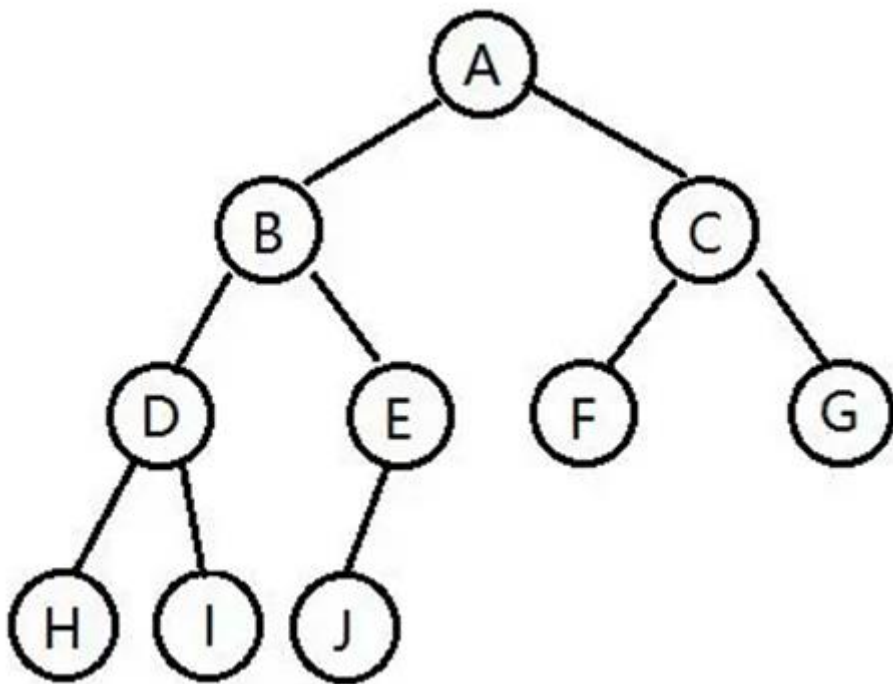
A. int B. float C. double D. 数组

【解析】数组一般不写在递归函数内部, 容易爆栈。

5. 一棵完全二叉树, 共有 1234 个节点, 其叶子结点的个数为 (C)。

A. 210 B. 618 C. 617 D. 512

【解析】第 i 层有 $2^{(i-1)}$ 个结点, 满二叉树节点总数 $2^n - 1$, 完全二叉树结点数 1234, $2^{11} = 2048$ 个结点, $2^{10} = 1024$, 确认高度 11 层, 第 11 层节点数 $1234 - 1024 = 210$ 。最后一层有 210 个叶结点, 210 个叶节点在倒数第二层有 105 个父节点。第 10 层有 $2^{(10-1)}$, 512 个节点, $512 - 105 + 210 = 617$ 个叶结点。



6. 某公司派赵钱孙李周五人出国学习，选派条件是：

- a. 若赵去，钱也去； b. 李、周两人必有一人去；
- c. 若周去，则赵、钱也同去； d. 孙、李二人同去或同不去；

如何选他们出国？（ B ）

- A. 孙赵周去 B. 赵钱周去 C. 李周孙去 D. 钱孙去

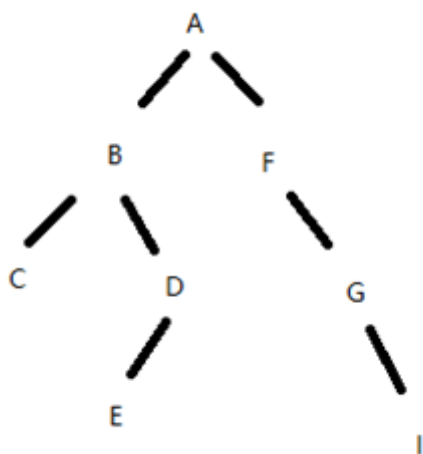
【解析】a、赵去钱去 c、周去 d、孙李二人同不去。所以赵钱周去符合条件。

7. 已知一棵二叉树前序遍历为 ABCDEFGI，后序遍历为 CEDBIGFA，则其中序遍历可能为（ B ）。

- A. ABCDEFGI B. CBEDAFIG C. CBDEAGFI D. CBEDAIFG

【解析】A 选项 先序 ABCDEFGI 中序 ABCDEFGI 画出二叉树，后续遍历不可能是 CEDBIGFA

B 选项 先序 ABCDEFGI 中序 CBEDAFIG



后序遍历 CEDBIGFA, 因此选 B

8. 8 颗子弹, 编号为 1, 2, 3, 4, 5, 6, 7, 8, 从编号 1 开始按序嵌入弹夹, 以下不是正常的打出子弹的次序的是 (D)。

A. 12345678 B. 87654321 C. 32154876 D. 32164587

【解析】子弹压入弹夹和打出的过程就是栈的操作过程, 本题就是按照 1, 2, 3, 4, 5, 6, 7, 8 顺序入栈, 找一个不合理的出栈顺序。A, B 都正确, C 是 1 入栈, 2 入栈, 3 入栈, 3 出栈, 2 出栈, 1 出栈, 4 入栈, 5 入栈, 5 出栈, 4 出栈, 6 入栈, 7 入栈, 8 入栈, 8 出栈, 7 出栈, 6 出栈。C 也可以。

D 1 入栈 2 入栈 3 入栈 3 出栈 2 出栈 1 出栈 4 入栈 5 入栈 6 入栈 6 出栈 (应该是 5 出栈) 所以 D 不合理。

9. 已知循环队列空间为 30, 队头位置编号为 12, 队尾元素下一个空位置编号为 5, 则队伍中元素个数为 (B)。

A. 22 B. 23 C. 7 D. 8

【解析】队头一般就是元素的开始存放位置, 队尾一般是元素存放位置的后一个空位置, 由题意可知, 元素是编号 12 开始, 编号 4 结束, 4 到 12 有 5, 6, 7, 8, 9, 10, 11 是空位置, 一共 7 个, 则 $30 - 7 = 23$, 所以选 B

10. 小童包中有 200 个笔杆, 其中有 160 个 X 型圆珠笔杆; 小美包中有 240 个笔芯, 其中有 180 个 X 型圆珠笔芯。现从两个包中各任取一个, 则能配成 X 型圆珠笔的概率为多少? (C)

A. $1/15$ B. $4/5$ C. $3/5$ D. $13/15$

【解析】 $(160/200) * (180/240) = 3/5$, 本题考察独立事件的概率计算, 公式是 $P(AB) = P(A) * P(B)$, A、B 是两个独立事件。独立事件是事件 B 发生或不发生对事件 A 不产生影响, 就说事件 A 与事件 B 之间存在某种“独立性”, 其对象可以是多个。小童包中取出 X 型圆珠笔杆的概率 $160/200$, 小美包中 X 型圆珠笔芯的概率是 $180/240$ 。两个包是两个独立事件, 配成一对的概率就是 $(160/200) * (180/240) = 3/5$ 。

11. 6 名学生, 老师将他们排成一圈分发奖品, 一共有 (B) 种排法。

A. 60 B. 120 C. 180 D. 240

【解析】

环排问题：n个人选择m个元素围成一圈，可以看成是m个顶点围成一圈，可以转化成直排问题去考虑，在相邻两个点去剪断，那么会拆分成m个直排。

直排：n个元素中选择m个元素排列，记作 A_n^m

环排：n个元素中选择m个元素围成一圈，记作N。对应的直排数量m*N：

所以直排数量 $m*N = A_n^m$ ，环排数量 $N = \frac{1}{m} * A_n^m$

回到本题：n个元素环排 $N = \frac{1}{n} * A_n^n = A_{n-1}^{n-1} = (n-1)!$

所以结果 $(6-1)! = 120$ 。

12. 设二维数组A的行下标为0至5，列下标为1至5，A的每个数据元素均占2个字节。在按行存贮的情况下，已知数据元素A[3][3]的第一个字节是2019，则A[4][4]的第一个字节的地址为（ D ）。

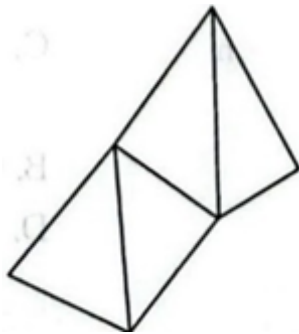
A. 2029 B. 2025 C. 2027 D. 2031

【解析】A[3][3]是2019，每个元素2个字节，那么

$A[3][4] = 2019 + 2 = 2021$, $A[3][5] = 2023$, $A[4][1] = 2025$

, $A[4][2] = 2027$, $A[4][3] = 2029$, $A[4][4] = 2031$ 。

13. 在下图中，有（ B ）个顶点出发存在一条路径可以遍历图中的每条边，而且仅遍历一次。



A. 6 B. 2 C. 3 D. 4

【解析】欧拉路径的条件，1、图是联通的。2、有且只有2个奇点。选B

14. 有A, B, C, D, E, F六个绝顶聪明又势均力敌的盗墓贼，他们都排着队，他们每个人都想独吞财宝，最前面的A如果拿了财宝，那么体力下降，则其后面的B会杀掉A，拿了财宝，当然B拿了财宝，体力也会下降，一样会被C杀掉，如果B不拿财宝，则C无法杀B，若每个人杀人必拿财宝，且优先保命，请问A、C、E的最终想法是（ C ）。

A. A不拿 C不拿 E拿 B. A拿 C拿 E不拿
C. A不拿 C不拿 E不拿 D. A不拿 C拿 E拿

【解析】F 后面无人了，所以 F 的想法是只要 E 拿了财宝就必杀 E，所以 E 不敢拿财宝，而 D 知道 E 不敢拿财宝，则一定会杀掉 C 拿财宝，所以 C 就不敢拿，而 B 知道 C 不敢拿，则 B 就会杀掉 A 拿财宝，所以 A 是不敢拿的，因此 ACE 三人的选择都是不拿，故选 C。

15. 以下不属于应用层的是（ D ）。

A. HTTP B. FTP C. TELNET D. UDP

【解析】TCP 和 UDP 属于传输层。

阅读程序

(1)

```
1  #include <bits/stdc++.h>
2  using namespace std;
3  int main() {
4      string s;
5      char s1[100];
6      int len, j=0;
7      cin>>s;
8      len=s.size();
9      memset(s1, 0, sizeof(s1));
10     for(int i=0; i<len; i++) {
11         if(i%2==0)
12             if((s[i]>='A' && s[i]<'Z') || (s[i]>='a' && s[i]<'z')) {
13                 s1[j]=s[i]+1;
14                 ++j;
15             }
16     }
17     cout<<s1<<endl;
18     return 0;
19 }
```

(1) 输出的字符串只能是字母组成。（ ）

答案 ✓

从 0 开始取偶数个字符并且在 A~(Z-1)，a~(z-1) 之间，+1 以后存到 s1，所以输出的只能是字母。

(2) 若将 12 行的 “<” 改为 “<=”，则输出结果有可能包含数字。（ ）

答案 ✕

Z 和 z +1 以后的字符不可能是数字，数字在 48~57 之间。

(3) 将第 9 行删除，程序运行结果不会改变。（ ）

答案 ✕

【解析】不初始化可能会有其他字符输出。

(4) 若将 11 行删除，则输出字符串的长度和输入字符串的长度一致。（ ）

答案 ✕

【解析】因为只有字母才会存入 s1 串，所以不一定长度一致。

(5) 若输入的字符串长度为 10，则输出的字符串长度最长可能为（ B ）。

A. 4 B. 5 C. 6 D. 10

【解析】长度为10，即下标（0~9）%2后全是字母的话就是0 2 4 6 8 五个字母。

（6）（4分）若输入的字符串都是字母，则输出中可能出现（ B ）。

A. A B. Z C. a D. 以上都不对

【解析】因为都是字母经过+1后，A和a是不可能出现的，Z是有可能的，输入偶数的字母为Y，+1后就是Z。

（2）

```
1  #include <bits/stdc++.h>
2  using namespace std;
3  int main() {
4      int a[1001], i, j, t, n;
5      for(i=0; i<=1000; i++)
6          a[i]=0;
7      scanf("%d", &n);
8      for(i=1; i<=n; i++) {
9          scanf("%d", &t);
10         a[t]++;
11     }
12     for(i=1000; i>=0; i--)
13         for(j=1; j<=a[i]; j++)
14             printf("%d ", i);
15     return 0;
16 }
```

（1）输入10个数字，输出结果是从小到大。（ ）

答案 ×

【解析】从大到小排序

（2）若输入的数字中有两个1，则输出时出来第一个1是第一个输入的。（ ）

答案 ×

【解析】无法确定是第几个输入的，因为a数组只记录了数量。

（3）（2分）若将第13行的“<=”改为“<”，且输入数据为10 2 12 33 34 1 2 87 22 9 0，则输出2。（ ）

答案 ✓

因为只有2出现了两次，改成小于的话出现一次的数字都不会输出。

（4）（2分）若将第12行改为for(i=0; i<=1000; i++)，则程序运行结果不变。（ ）

答案 ×

【解析】从小到大输出了

（5）若将第12行改为 for(i=1000; i>1; i--); 第13行改为 for(j=a[i]; j>1; j--), 输入数据为5 2 12 33 34 44, 则运行结果（ C ）。

A. 不变 B. 输出2 12 33 34 44 C. 无输出 D. 输出44 34 33 12 2

【解析】无输出，因为没有超过2个相同的数字，for(j=a[i]; j>1; j--), 1个数字的都不输出。

（6）（4分）若将第10行改为++a[t]或a[t++]，则输入5 1 2 3 4 5，输出结果为（ D ）。

- A. 1 2 3 4 5 或 5 4 3 2 1 B. 1 2 3 4 5 或无输出
C. 5 4 3 2 1 或 5 4 3 2 1 D. 5 4 3 2 1 或无输出

【解析】改成++a[t]没有变化还是由大到小输出，改成 a[t++]没有计数也没有其他操作，因此无输出。

(3)

```
1  #include <bits/stdc++.h>
2  using namespace std;
3  const int maxn=500000, INF=0x3f3f3f3f;
4  int L[maxn/2+2], R[maxn/2+2];
5  void unknown(int a[], int n, int left, int mid, int right) {
6      int n1=mid-left, n2=right-mid;
7      for(int i=0; i<n1; i++)
8          L[i]=a[left+i];
9      for(int i=0; i<n2; i++)
10         R[i]=a[mid+i];
11     L[n1]=R[n2]=INF;
12     int i=0, j=0;
13     for(int k=left; k<right; k++) {
14         if(L[i]<=R[j])
15             a[k]=L[i++];
16         else
17             a[k]=R[j++];
18     }
19 }
20 void unknownsort(int a[], int n, int left, int right) {
21     if(left+1<right) {
22         int mid=(left+right)/2;
23         unknownsort(a, n, left, mid);
24         unknownsort(a, n, mid, right);
25         unknown(a, n, left, mid, right);
26     }
27 }
28 int main() {
29     int a[maxn], n;
30     cin>>n;
31     for(int i=0; i<n; i++) cin>>a[i];
32     unknownsort(a, n, 0, n);
33     for(int i=0; i<n; i++) {
34         if(i) cout<<" ";
35         cout<<a[i];
36     }
37     cout<<endl;
38     return 0;
39 }
```

(1) 将第 13 行的 “<” 改为 “<=” 不会改变运行结果。()

答案 ×

【解析】结果会发生改变，因为多读取了数组的一个元素，会输出随机数。

(2) 将第 21 行的 “<” 改为 “<=” 不会改变运行结果。()

答案 ×

【解析】肯定会改变结果

(3) 此类排序方法是高效的，但是不稳定。()

答案 ×

【解析】归并排序，高效并且稳定。

(4) 将第 4 行的 2 个 “+2” 都去掉不会改变运行结果。()

答案 ✓

【解析】没用到那两个元素。

(5) 此题是哪种排序？(C)

A. 选择排序 B. 桶排序 C. 归并排序 D. 堆排序

【解析】归并排序

(6) (4 分) 此题用到了 (B) 思想。

A. 动态规划 B. 分治 C. 冒泡 D. 贪心

【解析】用到了分治的思想。

完善程序

1. 田忌赛马，田忌每赢一次齐王的马就得 200 金币，当然输了就扣 200 金币，平局则金币数不变。

```
1  #include <bits/stdc++.h>
2  using namespace std;
3  int main() {
4      int n;
5      while(cin>>n&&n!=0) {
6          int tj[1001],king[1001],count=0;
7          int tj_min=0,tj_max=n-1;
8          int king_min=0,king_max=n-1;
9          for(int i=0;i<n;i++) cin>>tj[i];
10         for(int i=0;i<n;i++) cin>>king[i];
11         sort(tj,tj+n);
12         sort(king,king+n);
13         while(n--){
14             if(tj[①]>king[②]){
15                 count++;
16                 tj_max--;
17                 king_max--;
18             }
19             else if(tj[③]<king[④]){
20                 count--;
21                 tj_min++;
22                 king_max--;
23             }
24             else
```



```

25         {
26             if(tj[tj_min]>king[king_min]){
27                 count++;
28                 ⑤;
29                 ⑥;
30             }
31             else{
32                 if( ⑦ )
33                     count--;
34                 tj_min++;
35                 ⑧;
36             }
37         }
38     }
39     cout<<count*200<<endl;
40 }
41 return 0;
42 }

```

此题属于贪心算法，思路就是田忌最快的比齐王最快的快，则和他比。田忌最慢的比齐王最慢的慢，则和齐王最快的比，反正要输，就把齐王最快的浪费掉。

(1) ①处和②处填 (A)。

- A. tj_max 和 king_max B. tj_min 和 king_max
C. tj_min 和 king_max D. tj_max 和 king_min

【解析】田忌最快的比齐王最快的快。

(2) ③处和④处填 (C)。

- A. tj_min 和 king_max B. tj_min 和 king_min
C. tj_max 和 king_max D. tj_max 和 king_min

【解析】田忌最快的比齐王最快的慢。

(3) ⑤处和⑥处填 (C)。

- A. tj_min--和 king_min++ B. tj_max++和 king_min++
C. tj_min++和 king_min++ D. tj_max++和 king_min--

【解析】田忌最慢的和齐王最慢的比，能赢则比，无法赢则找齐王最快的比。

(4) ⑦处填 (A)。

- A. tj[tj_min]<king[king_max] B. tj[tj_min]>king[king_max]
C. tj[tj_max]<king[king_max] D. tj[tj_min]>king[king_min]

【解析】田忌最慢的和齐王最快的比，输给最快的。

(5) ⑧处填 (A)。

- A. king_max-- B. king_max++
C. king_min-- D. king_min++

【解析】浪费掉齐王最快的，输掉一场。

2. 寻路问题: $N \times N$ 矩阵, 其中 0 是表示可以走的, 1 表示无法走, 矩阵由二维数组表示, 左上角是入口, 右下角是出口, 只能横着走和竖着走, 要求找出最短路径。

```
1  #include <bits/stdc++.h>
2  using namespace std;
3  int mymax=10000;
4  int f[4][2]={{-1,0},{1,0},{0,-1},{0,1}};
5  int a[20][20],v[20][20],v1[20][20];
6  int l=1;
7  int n;//矩阵的规模
8  bool check(int x1,int y1){
9      if(x1<0||x1>=n||    ①    ) return false;
10     if(a[x1][y1]==1||    ②    ) return false;
11     return true;
12 }
13 void dfs(int x,int y) {
14     if(x==n-1&&y==n-1){
15         if(l<mymax){
16             mymax=l;
17             memcpy(v1,v,sizeof(v1));
18         }
19         return;
20     }
21     for(int k=0;k<4;k++){
22         int x1,y1;
23         x1=x+    ③    ;
24         y1=y+    ④    ;
25         if(check(x1,y1)){
26             ⑤    ;
27             ⑥    ;
28             dfs(x1,y1);
29             ⑦    ;
30             v[x1][y1]=0;
31         }
32     }
33 }
34 int main(){
35     cin>>n;
36     for(int i=0;i<n;i++){
37         for(int j=0;j<n;j++){
38             cin>>a[i][j];
39         }
40         dfs(0,0);
41         int d=v1[n-1][n-1];
42         int x=n-1,y=n-1;
43         int k;
44         int qn[400][2];
45         qn[0][0]=n-1;
46         qn[0][1]=n-1;
47         for(k=1;;k++){
48             x=x-f[k][0];
```

```

49         y=y-f[d][1];
50         qn[k][0]=x;
51         qn[k][1]=y;
52         d=v1[x][y];
53         if(x==0&&y==0) break;
54     }
55     for(int i=k;i>=0;i--)
56         cout<< ⑧ <<" "<< ⑨ <<endl;
57     return 0;
58 }

```

(1)①处和②处填 (B)。

- A. $y1 \leq 0 \mid y1 > n$ 和 $v[x1][y1] > 0$ B. $y1 < 0 \mid y1 \geq n$ 和 $v[x1][y1] > 0$
 C. $y1 > 0 \&\& y1 \leq n$ 和 $v[x1][y1] > 0$ D. $y1 > 0 \&\& y1 < n$ 和 $v[x1][y1] > 0$

【解析】①是判断是否越界。②是判断是否有其他路可以走。

(2)③和④处填 (A)。

- A. $f[k][0]$ 和 $f[k][1]$ B. $f[k][1]$ 和 $f[k][0]$ C. $f[0][k]$ 和 $f[1][k]$ D. $f[1][k]$ 和 $f[0][k]$

【解析】计算下一步走的点。

(3)⑤处填 (B)。

- A. $v[x1][y1]=k+1$; B. $v[x1][y1]=k$; C. $v[x][y]=k$; D. $v[x][y]=k+1$;

解析：标记已走过,并且标记了走的下标。根据 48, 49 行上下文来判断。

(4)⑥处和⑦处填 (A)。

- A. $l++$ 和 $l--$ B. $k++$ 和 $k--$ C. $x1++$ 和 $x1--$ D. $y1++$ 和 $y1--$

【解析】 l 变量记录步骤数。走不通也要回溯。

(5)⑧处和⑨处填 (B)。

- A. $qn[i][1]$ 和 $qn[i][2]$ B. $qn[i][0]$ 和 $qn[i][1]$ C. $qn[1][i]$ 和 $qn[2][i]$ D. $qn[0][i]$
 和 $qn[1][i]$

【解析】

50 行 $qn[k][0]=x$;

51 行 $qn[k][1]=y$;

由这两行得到。