

某 NOIP 2018初赛 模拟赛

October 10, 2018

1 单项选择题（共 15 题，每题 1.5 分，共计 22.5 分。每题有且仅有一个正确选项）

1. 下列各数中为十进制 -0.625 的二进制补码表示的选项是（）

- A. 1.011 B. 0.011 C. 1.110 D. 0.110

2. 在 noi linux 系统下，如果你想要删库（所有文件），你需要在终端键入以下哪个指令（）

- A. `rm -r *` B. `rm -i *` C. `rm -f *` D. `rm -rf *`

3. 若一台计算机的字长为 64 位，则表明该计算机（）

- A. 在 CPU 中寄存器为 64 位 B. 能处理的数值最多由 8 个字节组成
C. 系统总线的带宽为 8 位 D. CPU 中运算的最大结果为 2^{64}

4. 下列有关 ENIAC 的信息正确的是（）

- A. 它于 1947 年 2 月 14 日诞生
B. 它运用了大量的晶体二极管和电子管
C. 它是世界上第一台电子计算机
D. 它运用了图灵提出的“存储程序通用电子计算机方案”方案

5. 2018 年 7 月 12 号是周四，那么 2029 年 12 月 7 号是（）

- A. 周一 B. 周三 C. 周五 D. 周日

6. 若某算法的计算时间表示为递推关系式：

$$T(n) = 8T(n/4) + n^{\frac{3}{2}}$$

$$T(1) = 1$$

则该算法的时间复杂度是（）

- A. $\Theta(n \log^3 n)$ B. $\Theta(n^2)$ C. $\Theta(n^2 \log n)$ D. $\Theta(n^{\frac{3}{2}} \log n)$

7. 由五个点构成的有根无标号二叉树个数是（）

- A. 42 B. 43 C. 44 D. 45

8.三进制数 2001 和 1222 异或的结果 ()

- A. 220 B. 2002 C. 2011 D. 2101

9.小S去古玩市场买了一个地址总线宽度为36的CPU: Pentium Pro, 那么它可以直接访问的物理空间有多大呢 ()

- A. 64 GB B. 16 GB C. 32 GB D. 8 GB

10.对于快速排序, 以下说法正确的是 ()

- A. 每一趟选定的基准元素到达有序时它应该在的位置
B. 每一趟选定的基准元素到达中间位置
C. 它主要运用了贪心的思想
D. 它主要运用了递归的思想

11.小S有 5 只彩色的互不相同的手机在盒子里, 她在跟某人玩游戏, 第 i 次她会放 i 只互不相同的手机 (与之前也不相同) 放进去, 然后某人拿出 i 只手机, 总共玩了 10 轮, 每一轮的分数为这轮不同的取出集合的数量, 那么 10 轮共能获得几分呢 ()

- A. 8849 B. 8047 C. 8007 D. 8809

12.小S对于接下来 10 期彩票各买了 2 张彩票, 每一期都总共放出 100 张彩票, 其中会有 1 张获得 100\$, 那么她想知道对于她这些彩票能赢得的总钱数的方差 ()

- A. 198\$ B. 1980\$ C. 196\$ D. 1960\$

13.下列有关图的说法正确的是 ()

- A. 无向联通图中对于任意两个点之间有且仅有一条最短路径
B. 若其为有向图且每个点只有一条出边, 那么构成了一颗基环外向树森林
C. 若一张无向图是欧拉图, 那么它没有一个奇数度数的节点, 反之亦然
D. 若图 $G = \langle V, E \rangle$ 是哈密顿图, 那么对于 V 的任意一个非空子集 S , 则对于删除 S 后的导出子图 $G - S$, 其联通分支数 $W(G - S)$, 满足 $W(G - S) \leq |S|$, 反之亦然

14. $P = false, Q = true, R = false, S = false$, 则下列逻辑表达式为真的是 ()

- A. $P \vee Q \wedge \neg R \rightarrow S$ B. $P \vee R \rightarrow S \rightarrow \neg Q$
C. $Q \rightarrow P \wedge R \wedge S$ D. $\neg(S \vee Q \rightarrow P) \wedge R$

15.下列有关noi相关事项的说法正确的是 ()

- A. 你可以在2021年的noi赛场上使用C语言
B. 你可以在2022年的noip赛场上使用pascal语言
C. 2018是CCF首次组队参加ISIJ2018
D. NOIP2017中浙江赛区因为程序收集丢失而扣除 1 个省队名额

2 二、不定项选择题 (共 5 题, 每题 1.5 分, 共计 7.5 分。每题有一个或多个正确选项, 多选或少选均不得分)

1. 下列选项正确的是 ()

- A. D不是正确的选项 B. D为正确选项

C. A为正确选项 D. B为错误选项

2. 下列为Google公司产品的由 ()

A. Github B. Blogger C. 王选 D. Hangouts

3. (不会ipv6的)Selina有一个局域网,其包含3个子网,其大小分别为[16384, 49152, 2147418112],子网掩码可能分别为(无序) ()

A. [210.33.19.103, 210.33.19.233, 210.33.19.520]

B. [255.255.0.0, 255.255.192.0, 128.0.0.0]

C. [255.255.255.0, 255.255.255.128, 255.255.192.0]

D. [192.0.0.0, 255.255.128, 255.255.255.0]

4. 下列关于凸包的说法正确的是 ()

A. 平面 $[1, n]^2$ 上的任意点集的凸包顶点个数最多为 $\Theta(n^{\frac{3}{4}})$

B. 人们可以使用物理引擎解决凸包问题

C. 求凸包直径(最远点对)理论最优时间复杂度为 $\Theta(n \log n)$

D. 关于 x, y, z, w 的方程 $x^4 + y^4 + z^4 = w^4$ 没有正整数解

5. 下列说法不正确的是 ()

A. 210.33.19.103为本地局域网

B. UDP协议在OSI中位于会话层

C. OSI模型有4层

D. 王选一度是美国国家科学院院士

3 三、问题求解 (共 2 题, 每题 5 分, 共计 10 分)

1. 小S有一块硬币,还有一个自然的头脑,维护了一个变量 A (初始值为 0), 每次他会抛起一个均匀的硬币, 如果结果为正面 $A = A + 1$, 否则 $A = \max(A - 1, 0)$, DolaBMOon 想要问你, A 在期望意义下将在雪莱抛()次硬币后达到1, ()次后达到 3, ()次后达到 2333。

2. 由 3 条Z形线所定义的区域的最大个数是(), 每条Z形线由两条平行的无限半直线和一条直线段组成。

4 四、阅读程序写结果 (共 4 题, 每题 8 分, 共计 32 分)

Hint: 我们保证程序能够正常运行不RE

4.1

```
#include<bits/stdc++.h>
```

```
using namespace std;
```

```

#define signed unsigned
#define QAQ signed

long long n;

int main(void) {
    cin >> n;
    assert(n == (1LL << 31));
    QAQ x = -233;
    cout << x << endl;
    return 0;
}

```

Input: 2147483648

Output:

4.2

```

#include <bits/stdc++.h>
using namespace std;

int n, K, x[101000];

int main(void) {
    scanf("%d%d", &n, &K);
    for(int i = 1; i <= n; i++)
        scanf("%d", &x[i]);
    int ans = 1e9;
    for(int i = 1; i <= n - K + 1; i++) {
        int j = i + K - 1;
        if(x[i] >= 0) ans = min(ans, x[j]);
        else if(x[j] <= 0) ans = min(ans, -x[i]);
        else ans = min(ans, min(-x[i] + x[j] - x[i], x[j] + x[j] - x[i]));
    }
    cout << ans << endl;
}

```

Input: 8 5 -9 -7 -4 -3 1 2 3 4

Output:

4.3

```

#include<bits/stdc++.h>

```

```

using namespace std;

const double eps=1e-6;

int Song(int x) {
    double ans=x,pre=0;
    while(abs(ans-pre)>=eps) {
        pre=ans;
        ans=(ans+x/ans)*0.5;
    }
    return ans;
}

int main() {
    cout<<Song(520);
    return 0;
}

```

Output:

4.4

```

#include <bits/stdc++.h>
using namespace std;

bool visr[12], visc[12];
int cnt[26], n, m, row[12], col[12];
string s[12];

bool EqualCol(int x, int y) {
    memset(cnt, 0, sizeof(cnt));
    for(int i = 0; i < n; i++)
        cnt[(int)(s[i][x] - 'a')]++,
        cnt[(int)(s[i][y] - 'a')]--;
    for(int i = 0; i < 26; i++)
        if(cnt[i]) return 0;
    return 1;
}

bool EqualRow(int x, int y) {
    memset(cnt, 0, sizeof(cnt));
    for(int i = 0; i < m; i++)
        cnt[(int)(s[x][i] - 'a')]++,
        cnt[(int)(s[y][i] - 'a')]--;
    for(int i = 0; i < 26; i++)

```

```

        if(cnt[i]) return 0;
    return 1;
}

bool Check(void) {
    for(int i = 0; i < n; i++)
        for(int j = 0; j < m; j++)
            if(s[row[i]][col[j]] != s[row[n - i - 1]][col[m - j - 1]]) return 0;
    return 1;
}

bool SolveCol(int x) {
    if(x == (m + 1) / 2) return Check();
    if((m & 1) && (x == m / 2)) {
        for(int i = 0; i < m; i++)
            if(!visc[i])
                col[x] = i;
        return SolveCol(x + 1);
    }
    for(int i = 0; i < m; i++)
        if(!visc[i])
            for(int j = i + 1; j < m; j++)
                if(!visc[j] && EqualCol(i, j)) {
                    visc[j] = visc[i] = 1;
                    col[x] = i;
                    col[m - x - 1] = j;
                    if(SolveCol(x + 1)) return 1;
                    visc[j] = visc[i] = 0;
                }
    return 0;
}

bool SolveRow(int x) {
    if(x == (n + 1) / 2) return SolveCol(0);
    if((n & 1) && (x == n / 2)) {
        for(int i = 0; i < n; i++)
            if(!visr[i])
                row[x] = i;
        return SolveRow(x + 1);
    }
    for(int i = 0; i < n; i++)
        if(!visr[i])
            for(int j = i + 1; j < n; j++)
                if(!visr[j] && EqualRow(i, j)) {
                    visr[i] = visr[j] = 1;
                    row[x] = i;
                }

```

```

        row[n - x - 1] = j;
        if(SolveRow(x + 1)) return 1;
        visr[i] = visr[j] = 0;
    }
    return 0;
}

int main(void) {
    scanf("%d%d", &n, &m);
    for(int i = 0; i < n; i++) {
        cin >> s[i];
    }
    puts(SolveRow(0) ? "YES" : "NO");
    return 0;
}

```

Input:2 3 abc cab

Output:

5 五、完善程序（共 2 题，其中第1，4个空2分，其余3分一空，共计 28 分）

5.1 A 的 $Moon$ 次方

输入 A ， $Moon$ ，输出 A^{Moon} 。

```

#include<bits/stdc++.h>

using namespace std;

#define int long long

const int MOD=1926081711*19260817;

int Mul(int x,int y) {
    int tmp=(x*y-(__1__)((long double)(__2__)+1.0e-8)*MOD);
    return __3__;
}

int Qpow(int x,int y=MOD-2) {
    int res=1;
    for(;y;y>>=1,x=Mul(x,x))if(__4__)
        res=Mul(res,x);
    return res;
}

```

```

signed main() {
    int A,Moon;
    cin>>A>>Moon;
    cout<<__5__<<endl;
    return 0;
}

```

5.2 加工生产调度

某工厂收到了 n 个产品的订单，这 n 个产品分别在 A 、 B 两个车间加工，并且必须先先在 A 车间加工后才可以到 B 车间加工。

某个产品 i 在 A 、 B 两车间加工的时间分别为 A_i 、 B_i 。怎样安排这 n 个产品的加工顺序，才能使总的加工时间最短。这里所说的加工时间是指：从开始加工第一个产品到最后所有的产品都已在 A 、 B 两车间加工完毕的时间。

```

#include<bits/stdc++.h>

using namespace std;

const int N=1111;

struct Data {
    __6__
}p[N];

int n;

bool Cmp(const Data &a,const Data &b) {
    return a.a+__7__<a.b;
}

int main() {
    scanf("%d",&n);
    for(int i=1;i<=n;++i)
        scanf("%d",&p[i].a);
    for(int i=1;i<=n;++i)
        scanf("%d",&p[i].b);
    for(int i=1;i<=n;++i)
        p[i].id=i;
    sort(__8__);
    int A=p[1].a,B=p[1].a+p[1].b;
    for(int i=2;i<=n;++i) {
        B=__9__;
        A+=p[i].a;
    }
}

```



```
}  
printf("%d\n",B);  
for(int i=1;i<=n;++i)  
    printf("%d ",__10__);  
return 0;  
}
```