启航舱暑假集训竞赛二班 8.8 模拟赛

一．选择题（共 15 题，每题 2 分，共 30 分）

1.二进制数

11 1011 1001 0111 和 01 0110 1110 1011 进行按位与运算的结果是（）

A. 01 0010 1000 1011

B. 01 0010 1001 0011

C. 01 0010 1000 0001

D. 01 0010 1000 0011

2.一个 32 位整型变量占用（）个字节。

A. 32　　B. 128　　C. 4　　D. 8

3.若有如下程序段，其中 s、a、b、c 均已定义为整型变量,且 a、c 均已赋值（c 大于 0）

```
s = a;
for （b = 1; b <= c; b++） s = s - 1;
```

则与上述程序段功能等价的赋值语句是（）

A. s = a - c;　　B. s = a - b;

C. s = s - c;　　D. s = b - c;

4.设有 100 个已排好序的数据元素，采用折半查找时，最大比较次数为（）

A. 7　　B. 10

C. 6　　D. 8

5.新学期开学了，小胖想减肥，健身教练给小胖制定了两个训练方案。

方案一：每次连续跑 3 公里可以消耗 300 千卡（耗时半小时）；

方案二：每次连续跑 5 公里可以消耗 600 千卡（耗时 1 小时）。

小胖每周周一到周四能抽出半小时跑步，周五到周日能抽出一小时跑步。

另外，教练建议小胖每周最多跑 21 公里，否则会损伤膝盖。

请问如果小胖想严格执行教练的训练方案，并且不想损伤膝盖，每周最多通过跑步消耗多少千卡？（）

A. 3000

B. 2500

C. 2400

D. 2520

6.以下哪个奖项是计算机科学领域的最高奖？（）

A. 图灵奖

B. 鲁班奖

C. 诺贝尔奖

D. 普利策奖

7.排序的算法很多，若按排序的稳定性和不稳定性分类，则（）是不稳定排序。

A. 冒泡排序
B. 插入排序
C. 选择排序
D. 计数排序

8.在内存储器中每个存储单元都被赋予一个唯一的序号，称为（）。
A. 地址
B. 序号
C. 下标
D. 编号

9.编译器的主要功能是（ ）。
A. 将源程序翻译成机器指令代码
B. 将源程序重新组合
C. 将低级语言翻译成高级语言
D. 将一种高级语言翻译成另一种高级语言

10.冒泡排序算法的伪代码如下：

```
输入：数组 L，n ≥ k。输出：按非递减顺序排序的 L。
算法 BubbleSort：
   1. FLAG ← n //标记被交换的最后元素位置
   2. while FLAG > 1 do
   3.     k ← FLAG -1
   4.     FLAG ← 1
   5.     for j=1 to k do
   6.         if L(j) > L(j+1) then do
   7.             L(j)  ↔ L(j+1)
   8.             FLAG ← j
```

对 n 个数用以上冒泡排序算法进行排序，最少需要比较多少次?（ ）。
A. n^2　　B. n-2　　C. n-1　　D. n

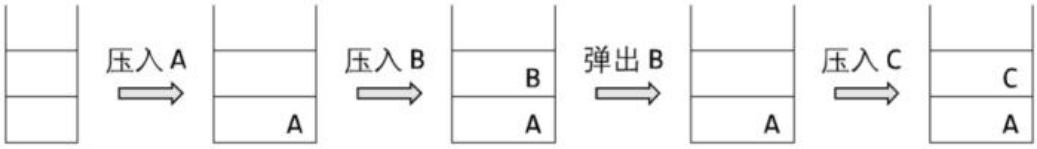
11.设 A 是 n 个实数的数组，考虑下面的递归算法：

```
XYZ (A[1..n])
1.  if n=1 then return A[1]
2.  else temp ← XYZ (A[1..n-1])
3.  if temp < A[n]
4.  then return temp
5.  else return A[n]
```

请问算法 XYZ 的输出是什么？（）。
A. A 数组的平均
B. A 数组的最小值
C. A 数组的中值
D. A 数组的最大值

12.二进制数 1011 转换成十进制数是（ ）。
A. 11
B. 10

C. 13
D. 12
13.下图中所使用的数据结构是（)。



A. 栈　　B. 队列　　C. 二叉树　　D. 哈希表

14.以下不属于面向对象程序设计语言的是（)。
A. C++　　B. Python　　C. Java　　D. C
15.对于入栈顺序为 a,b,c,d,e 的序列，下列（ ）不是合法的出栈序列。
A. a,b,c,d,e
B. e,d,c,b,a
C. b,a,c,d,e
D. c,d,a,e,b
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）

```
#include <cstdio>
#include <cstring>
using namespace std;
char st[100];
int main() {
    scanf("%s", st);
    int n = strlen(st);
    for (int i = 1; i <= n; ++i) {
        if (n % i == 0) {
            char c = st[i - 1];
            if (c >= 'a')
                st[i - 1] = c - 'a' + 'A';
        }
    }
    printf("%s", st);
    return 0;
}
```

判断题
1.输入的字符串只能由小写字母或大写字母组成。（）
2.若将第 8 行的 i = 1 改为 i = 0，程序运行时会发生错误。（）
3.若将第 8 行的 i <= n 改为 i * i <= n，程序运行结果不会改变。（）
4.若输入的字符串全部由大写字母组成，那么输出的字符串就跟输入的字符串一样。（）
选择题

5.若输入的字符串长度为 18，那么输入的字符串跟输出的字符串相比，至多有（）个字符不同。
6.若输入的字符串长度为(),那么输入的字符串跟输出的字符串相比,至多有 36 个字符不同。
1. A. 正确 B. 错误
2. A. 正确 B. 错误
3. A. 正确 B. 错误
4. A. 正确 B. 错误
5. A. 18 B. 6 C. 10 D. 1
6. A. 36 B. 100000 C. 1 D. 128

```cpp
#include<cstdio>
using namespace std;
int n, m;
int a[100], b[100];

int main() {
    scanf("%d%d", &n, &m);
    for (int i = 1; i <= n; ++i)
        a[i] = b[i] = 0;
    for (int i = 1; i <= m; ++i) {
        int x, y;
        scanf("%d%d", &x, &y);
        if (a[x] < y && b[y] < x) {
            if (a[x] > 0)
                b[a[x]] = 0;
            if (b[y] > 0)
                a[b[y]] = 0;
            a[x] = y;
            b[y] = x;
        }
    }
    int ans = 0;
    for (int i = 1; i <= n; ++i) {
        if (a[i] == 0)
            ++ans;
        if (b[i] == 0)
            ++ans;
    }
    printf("%d", ans);
    return 0;
}
```

假设输入的 n 和 m 都是正整数，x 和 y 都是在 [1,n] 的范围内的整数，完成

下面的判断题和单选题：
判断题
1.当 m>0 时，输出的值一定小于 2n。（）
2.执行完第 27 行的 ++ans 时，ans 一定是偶数。（）
3.a[i] 和 b[i] 不可能同时大于 0。（）
4.右程序执行到第 13 行时，x 总是小于 y，那么第 15 行不会被执行。（）
选择题
5.若 m 个 x 两两不同，且 m 个 y 两两不同，则输出的值为（）
6.若 m 个 x 两两不同，且 m 个 y 都相等，则输出的值为（）
1. A. 正确 B. 错误
2. A. 正确 B. 错误
3. A. 正确 B. 错误
4. A. 正确 B. 错误
5.A. 2n−2m B. 2n+2 C. 2n−2 D. 2n
6.A. 2n−2 B. 2n C. 2m D. 2n−2m

```
#include <iostream>
using namespace std;

long long n, ans;
int k, len;
long long d[1000000];

int main() {
  cin >> n >> k;
  d[0] = 0;
  len= 1;
  ans = 0;
  for (long long i = 0; i <n; ++i) {
    ++d[0];
    for (int j = 0; j + 1<len; ++j) {
      if (d[j] == k) {
        d[j] = 0;
        d[j + 1] += 1;
        ++ans;
      }
    }
    if (d[len- 1] == k) {
      d[len - 1] = 0;
      d[len] =1;
      ++len;
      ++ans;
    }
  }
```

```
  cout << ans << endl;
  return 0;
}
```

假设输入的 n 是不超过 2^62 的正整数，k 都是不超过 10000 的正整数，完成下面的判断题和单选题：

判断题

1.若 k=1，则输出 ans 时，len=n。（ ）

2.若 k>1，则输出 ans 时，len 一定小于 n。（ ）

3.若 k>1，则输出 ans 时，k^len一定大于 n。（ ）

单选题

4.若输入的 n 等于：10^15 ，输入的 k 为 1 则输出等于（ ）。

5.若输入的 n 等于 205,891,132,094,649（即 3^{30}），输入的 k 为 3，则输出等于（ ）

6.若输入的 n 等于 100,010,002,000,090，输入的 k 为 10，则输出等于（ ）。

1. A. 正确 B. 错误

2.A. 正确 B. 错误

3.A. 正确 B. 错误

4. A. 1　　B. $(10^{30} - 10^{15})/2$

C. $(10^{30} + 10^{15})/2$　　D. 10^{15}

5.

A. 3^{30}　　B. $(3^{30} - 1)/2$

C. $3^{30} - 1$　　D. $(3^{30} + 1)/2$

6.

A. 11,112,222,444,543

B. 11,122,222,444,453

C. 11,122,222,444,543

D. 11,112,222,444,453

三、完善程序（共两题，每题有 5 个选择题，共 30 分）

输入月份 m(1≤m≤12)，按一定格式打印 2015 年第 m 月的月历。（第三、四空 2.5 分， 其余 3 分）

例如，2015 年 1 月的月历打印效果如下（第一列为周日）：

S	M	T	W	T	F	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

```
01 #include <iostream>
02 #include <string>
```

```
03 using namespace std;
04 const int dayNum[] = {-1, 31, 28, 31, 30, 31, 30, 31, 31, 30,
31, 30, 31};
05 int m, offset, i;
06 int main(){
07     cin >> m;
08     cout << "S\tM\tT\tW\tT\tF\tS" << endl; /* '\t'为 TAB 制表
符 */
09     ①;
10     for (i = 1; i < m; i++)
11         offset = ②;
12     for (i = 0; i < offset; i++)
13         cout << '\t';
14     for (i = 1; i <= ③; i++)
15     {
16         cout << ④;
17         if (i == dayNum[m] || ⑤ == 0)
18             cout << endl;
19         else
20             cout << '\t';
21     }
22     return (0);
23 }
```

1. ①处应填（ ）

A. offset = 0　　B. offset = 4

C. offset = 6　　D. offset = 1

2. ②处应填（ ）

A. offset + dayNum[i]　　B. (offset + dayNum[m]) % 7

C. (offset + dayNum[i]) % 7　　D. dayNum[i] % 7

3. ③处应填（ ）

A. dayNum[m-1]　　B. dayNum[m]

C. m　　D. 31

4. ④处应填（ ）

A. offset　　B. dayNum[i]

C. i+1　　D. i

5. ⑤处应填（ ）

A. (offset + i) % 7　　B. (offset + i - 1) % 7

C. i % 7　　D. offset % 7

有 n 名同学参加学校组织的郊游活动，已知学校给这 n 名同学的郊游总经费为 A 元，与此同时第 i 位同学自己携带了 Mi 元。为了方便郊游，活动地点提供 B(>=n) 辆自行车供人租用，租用第 j 辆自行车的价格为 Cj 元，每位同学可以使用自己携带的钱或者学校的郊游经费，为了方便账务管理，每位同学只能为自己租用自行车，且不会借钱给他人，他们想知道最多有多少位同学能够租用到自行车。

本题采用二分法。对于区间[l, r]，我们取中间点 mid 并判断租用到自行车的人数能否达到 mid。判断的过程是利用贪心算法实现的。

```
01 #include <iostream>
02 using namespace std;
03 #define MAXN 1000000
04
05 int n, B, A, M[MAXN], C[MAXN], l, r, ans, mid;
06
07 bool check(int nn)
08 {
09     int count = 0, i, j;
10     i = ①;
11     j = 1;
12     while (i <= n)
13     {
14         if (②)
15             count += C[j] - M[i];
16         i++;
17         j++;
18     }
19     return ③;
20 }
21
22 void sort(int a[], int l, int r)
```

```
23 {
24     int i = l, j = r, x = a[(l + r) / 2], y;
25     while (i <= j)
26     {
27         while (a[i] < x) i++;
28         while (a[j] > x) j--;
29         if (i <= j)
30         {
31             y = a[i];
32             a[i] = a[j];
33             a[j] = y;
34             i++;
35             j--;
36         }
37     }
38     if (i < r) sort(a, i, r);
39     if (l < j) sort(a, l, j);
40 }
41
42 int main()
43 {
44     int i;
45     cin >> n >> B >> A;
46     for (i = 1; i <= n; i++)
47         cin >> M[i];
48     for (i = 1; i <= B; i++)
49         cin >> C[i];
50     sort(M, 1, n);
51     sort(C, 1, B);
52     l = 0;
53     r = n;
54     while (l <= r)
55     {
56         mid = (l + r) / 2;
```

```
57          if (④)
58          {
59              ans = mid;
60              l = mid + 1;
61          }
62          else
63              r = ⑤;
64      }
65      cout << ans << endl;
66      return 0;
67  }
```

1. ①处应填（ ）

A. 1　　B. n - nn + 1

C. nn　　D. n - nn

2. ②处应填（ ）

A. M[i] == C[j]　　B. M[i] > C[j]

C. M[i] < C[j]　　D. M[i] != C[j]

3. ③处应填（ ）

A. count >= A　　B. count <= A

C. count < A　　D. count > A

4. ④处应填（ ）

A. check(mid)　　B. !check(mid)

C. mid <= A　　D. mid >= A

5. ⑤处应填（ ）

A. 1　　B. mid

C. mid + 1　　D. mid - 1