

BBCC U18组 卷一 (样题)

一、选择题，共计100分。

1、一位摄影师爱好拍照，他有20000幅1024*768的真彩色图像，如果将这些图像以位图形式保存在CD光盘上（一张CD光盘的容量按照600M计算），大约需要（）张CD光盘。（5分）

A、10 B、50 C、100 D、1000

2、下列无符号数中，最小的数是（）。（5分）

A、(111101001)₂ B、(76)₁₀ C、(35)₈ D、(2A)₁₆

3.在衡量存储器容量时，计量单位由小到大的顺序是（）。（5分）

A、KB GB TB B、TB KB GB C、TB GB KB D、KB TB GB

4、下列IP地址中正确的是（）。（5分）

A、12.201.256.12 B、192.97.32.121 C、126,33,101,125 D、127:111:35:91

5、最优前缀编码，也称Huffman编码。这种编码组合的特点是对于较频繁使用的元素给与较短的唯一编码，以提高通讯的效率。下面编码组合哪一组不是合法的前缀编码：（）（5分）

A. (00, 01, 10, 11) B. (0, 1, 00, 11) C. (0, 10, 110, 111) D. (1, 01, 000, 001)

6、给定有权无向图的邻接矩阵如下，其最小生成树的总权重是：（）（5分）

$$\begin{bmatrix} \infty & 3 & 1 & 15 & 1 & 9 \\ 3 & \infty & 3 & 13 & \infty & 8 \\ 1 & 3 & \infty & 1 & 1 & 2 \\ 15 & 13 & 1 & \infty & 18 & \infty \\ 1 & \infty & 1 & 18 & \infty & 2 \\ 9 & 8 & 2 & \infty & 2 & \infty \end{bmatrix}$$

A、20 B、22 C、8 D、15

7、对有n个结点、m条边的无向图，采用邻接矩阵存储，现在进行深度优先遍历，则时间复杂度为（）。（5分）

A.O(m²) B.O(n+m) C.O(n*m) D.O(n²)

8、当a=9, b=2, c=3, d=10时后缀表达式abc+-db/-的运算结果为（）（5分）

A.9 B.1 C.-1 D.-9

- 9、若3个顶点的无权图G的邻接矩阵用数组存储为 $\{0, 1, 1\}\{1, 0, 1\}\{0, 1, 0\}$ ，假定在具体存储中顶点依次为： v_1, v_2, v_3 关于该图，下面的说法哪个是错误的： () (5分)
- A、该图是有向图。
 - B、该图是强联通的。
 - C、该图所有顶点的入度之和减所有顶点的出度之和等于1。
 - D、从 v_1 开始的深度优先遍历所经过的顶点序列与广度优先的顶点序列是相同的。

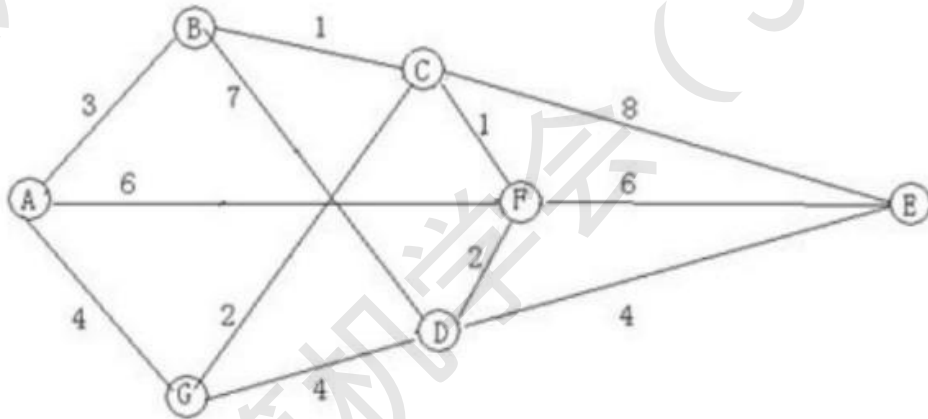
10、在一个不带头结点的非空链式队列中,假设f和r分别为队头和队尾指针,则插入s所指的结点运算是 ()。(5分)

- A、 $f \rightarrow next = s; f = s;$ B、 $r \rightarrow next = s; r = s;$ C、 $s \rightarrow next = s; r = s;$ D、 $s \rightarrow next = f; f = s;$

11、G是一个非连通简单无向图，现有55条边，则该图至少有 () 个顶点。(5分)

- A、12 B、13 C、14 D、15

12、如图所示，图中每条边上的数字表示该边的长度，则从A到E的最短距离是 ()。(5分)



- A、14 B、13 C、11 D、12

13、已知一棵二叉树的前序遍历结果为ABDECFHJIG，中序遍历的结果为DBEAJHFICG,则这棵二叉树的深度为 ()。(5分)

- A、4 B、5 C、6 D、7

14、某位NBA球员三分球命中率为40%，设N表示他在比赛中首次投中三分，累计已投三分的次数，计算N取奇数的概率 () 假定各次投篮是否命中相互独立。(5分)

- A、3/8 B、5/16 C、5/8 D、11/16

15、大湾区中学高三(1)班有50人，39人会游泳，36人会骑车，40人会溜冰，45人会打羽毛球，至少有 () 人四项都会。(5分)

- A、8 B、9 C、10 D、11

16、斐波那契数列为 1, 1, 2, 3, 5, 8, 13, 21, ... ,其元素产生的规则是前两个数为1, 第三个数开始每个数等于它前面两个数之和。已知任意一个正整数可以表示为若干个互不相同的斐波那契数之和。(10分)

例如: $36 = 21 + 13 + 2$ 。

下面的程序是由键盘输入一个正整数N, 输出组成N 的互不相同的斐波那契数。算法说明:

- ①寻找小于等于N的最大斐波那契数A, 并以A作为组成N的一个数;
- ②若 $N \neq A$, 则以 $N-A$ 组为N的新值, 重复步骤①。若 $A=N$, 则结束。

```
#include<iostream>
#include<iomanip>
using namespace
std; int N;
bool first;
int find(int
N){ int
A=1, B=1, C;
do{
C=____(1)____ ; A=B; B=C;
}while(B<N);
if(____(2)____) return ____ (3) ____ ; else return ____ (4) ____;
```

```
}
void p(){
int A=find(N);
if(first) {
cout<<A;
first=false;
}else
cout<<"+"<<A;
if(A<N) p(____(5)____);
}
int main(){
cin>>N;
first=true;
cout<<N<<"=";
p(N);
cout<<endl;
```

- A. $B=N$
- B. $N-A$
- C. $B-A$
- D. A
- E. N
- F. $A+B$

17、(15分) (棋盘覆盖问题) 在一个 $2^k \times 2^k$ 个方格组成的棋盘中恰有一个方格与其它方格不同 (图中标记为-1的方格), 称之为特殊方格。现用L型 (占3个小方格) 纸片覆盖棋盘上除特殊方格的所有部分, 各纸片不得重叠, 于是, 用到的纸片数恰好是 $(4^k - 1) / 3$ 。在下表给出的一个覆盖方案中, $k=2$, 相同的3个数字构成一个纸片。下面给出的程序使用分治法设计的, 将棋盘一分为四, 依次处理左上角、右上角、左下角、右下角, 递归进行。请将程序补充完整。

例:

```
2 2 3 3
2 -1 1 3
4 1 1 5
4 4 5 5
```

-1为特殊方格的位置, 其他数字表示L型方格放置的顺序。

【输入描述】

两行, 第一行一个整数size, 表示棋盘大小; 第二行两个整数, 表示特殊方格位置

【输出描述】

如题意, 输出棋盘。

```
#include <bits/stdc++.h>
using namespace std;
int board[65][65], tile; /* tile为纸片编号 */
void chessboard(int tr, int tc, int dr, int dc, int size){
    /* dr, dc依次为特殊方格的行、列号 */
    int t, s;
```

```

if (size == 1) return ;
t=tile++;
s=size/2;
if (____(1)____){
    chessboard( tr, tc, dr, dc, s);
}else{
    board[tr + s -1][tc + s -1] = t;
    ____ (2) ____;
}
if ( dr < tr + s && dc >= tc + s )
    chessboard( tr, tc + s, dr, dc, s);
else {
    board[tr + s -1][tc + s] = t;
    ____ (3) ____;
}
if ( dr >= tr + s && dc < tc + s )
    chessboard( tr + s, tc, dr, dc, s);
else {
    board[tr + s][tc + s -1] = t;
    ____ (4) ____;
}
if ( dr >= tr + s && dc >= tc + s )
    chessboard( tr + s, tc + s, dr, dc, s);
else {
    board[tr + s][tc + s] = t;
    ____ (5) ____;
}
}
void prt1(int b[][65],int
n){ int i,j;
for (i=1;i<=n;i++){
    for (j=1;j<=n;j++) cout << setw( 3 ) << b[i][j]; cout <<endl;
}
}
int main(){
int size,dr,dc;
cin>> size ;
cin>> dr >> dc ;
board[dr][dc] = -1;
tile++;
chessboard( 1, 1, dr, dc, size);
prt1( board, size );
return 0;
}

```

- A、 chessboard(tr,tc+s,tr+s-1,tc+s,s)
- B、 chessboard(tr,tc,tr+s-1,tc+s,s)
- C、 chessboard(tr+s,tc+s,tr+s,tc+s,s)
- D、 chessboard(tr,tc,tr+s-1,tc+s-1,s)
- E、 (dr<tr+s)&&(dc<tc+s)
- F、 chessboard(tr+s,tc,tr+s,tc+s-1,s)

BBCC U18组 卷二 (样题)

二、程序设计题 (200分) : 共2题, 每题100分

1、删除游戏

文件名	输入文件	输出文件	时间限制	空间限制
delnum.cpp	delnum.in	delnum.out	1s	512MB

题目描述

因为疫情影响, 小粤和小港周末只能呆在家里玩, 小粤有一个好的想法, 他现在的手上有 n 个数字, 编号从 a_1 到 a_n , 要求小港删除其中的 k 个数字, 使得删除相应的数字之后, 剩下的 $n - k$ 个数中不相同的数字最多, 小港冥思苦想也没有想出来, 你能帮他想对策么?

输入格式

第一行输入两个正整数 n 和 k 。

第二行有 n 个非负整数, 分别代表 a_1 到 a_n 这些数字。

输出格式

一行数字, 一个整数result, 表示删除了 k 个数字后最多的不同的数的个数。

样例输入

```
4 1
1 3 1 2
```

输出样例

```
3
```

数据范围

对于 30% 的数据, $n \leq 10, a_i \leq 10$.

对于 60% 的数据, $n \leq 100, a_i \leq 100$.

对于 80% 的数据, $n \leq 10^5, a_i \leq 10^5$.

对于 100% 的数据, $n \leq 10^5, a_i \leq 10^9$.

2、密室逃脱

文件名	输入文件	输出文件	时间限制	空间限制
escape.cpp	escape.in	escape.out	2s	512MB

小J被关在密室里!

密室的构造是一个 $H \times W$ 的棋盘, 即有 **H**行**W**列, 第 i 行第 j 列的房间用 $A_{i,j}$

- 若 $A_{i,j} = \#$, 则该房间的门是锁上的;
- 若 $A_{i,j} = \cdot$, 则该房间是可以自由进出的;

小J被困在 $A_{i,j} = \text{'S'}$ 的地方, 该房间是可以自由进出的。

房间是四相邻的, 即 (i, j) 可以移动到 $(i + 1, j), (i - 1, j), (i, j + 1), (i, j - 1)$

1) 小J将通过如下方式逃脱密室, 每一回合, 她会按顺序执行如下操作:

- 移动至多 K 次, 可以不移动, 此时被锁着的房间是不能够进入的
- 选择至多 K 个锁着的房间打开, 可以不选择任何房间, 从此那些房间将一直可以自由进出

出口在密室的四周, 即第1行, 第1列, 第 H 行, 第 W 列的所有房间, 出口一开始可能是被锁住的。你需要用最少的回合移动到出口, 输出这个回合数。

输入格式

第一行读入三个整数 H, W, K

接下来 H 行, 每行 W 个字符, 表示 $A_{i,j}$

输出格式

输出一行一个数, 表示小 J 最少要用多少回合逃出密室。

样例

样例输入 #1

```
3 3 3
#.#
#S.
###
```

样例输出 #1

```
1
```

样例解释:

小J可以在一个回合内移动到 (1 , 2)

样例输入 #2

```
3 3 3
###
#S#
###
```

样例输出 #2

```
2
```

样例解释:

小J第一个回合不能移动,但她解锁了房间 (1 , 2), 然后她在第二个回合移动到 (1 , 2) 逃离密室。

样例输入 #3

```
7 7 2
#####
#####
##...##
###S###
##.#.##
###.###
#####
```

样例输出 #3

```
2
```

数据范围与提示

$3 \leq H, W \leq 800$

$1 \leq K \leq H \times W$

$A_{i,j}$ 是 '#', '.', 'S' 中的一种, 有且仅有一个 $A_{i,j}$ 为 'S'

Subtask1 (30分): $H, W \leq 4$

Subtask2 (10分): 不存在 $A_{i,j} = \#$

Subtask3 (10分): 不存在 $A_{i,j} = \cdot$

Subtask3 (50分): 无特殊限制