

1. Dane są deklaracje reprezentujące szachownicę o boku długości N:

```
const int N= ...  
int tab[N][N];
```

Tablica tab jest wypełniona liczbami naturalnymi. Na szachownicy umieszczamy dwa klocki domina tak, że jeden klocek przykrywa dwa pola. Proszę napisać funkcję, która sprawdza czy istnieje takie ustawianie klocków na szachownicy, że:

- oba klocki są prostopadłe do siebie,
- w żadnym wierszu ani w żadnej kolumnie nie leży więcej niż jeden klocek,
- największym wspólnym dzielnikiem 4 przykrytych liczb jest jeden.

2. W grze mag-mino wykorzystuje się klocki, które mają kształt prostokątów, na których obydwu końcach znajduje się liczba oczek od 0 do 9. Na każdym klocku z dwóch jego końców liczba oczek jest inna. W komplecie liczącym 90 klocków do gry występują wszystkie kombinacje oczek i każda kombinacja występuje dokładnie jeden raz. Proszę napisać funkcję, która dla danego zbioru N klocków wyznacza najdłuższy ciąg jaki można z nich ułożyć.

Na przykład dla zbioru 8 klocków: [2|8] [0|1] [2|3] [3|6] [2|6] [2|9] [3|4] [6|7]

najdłuższy ciąg jaki można ułożyć ma długość 5 i ma postać : [8|2] [2|3] [3|6] [6|2] [2|9]

Dane opisujące zestaw:

```
const int N= ...
struct klocek {
    int a;
    int b; // b>a
};

klocek zestaw[N];
```

Do funkcji należy przekazać zestaw klocków, funkcja powinna zwrócić największą długość ciągu jaki można z tego zestawu zbudować. Wskazówka : kiedy z zestawu klocków da się zbudować ciąg?

3. Dane są deklarację reprezentujące listę z klockami mag-mina (patrz zadanie 2).

```
struct klocek {  
    int a;  
    int b;  
    klocek *next;  
};
```

Lista zawiera zestaw klocków, które można ułożyć w ciąg. Niestety klocki pomieszały się. Proszę napisać funkcję, która ustawia klocki na liście w ciąg. Uwaga: orientacja klocków w liście jest właściwa.

Na przykład listę: [2|9] [3|6] [8|2] [2|3] [6|2]

należy przekształcić na listę: [8|2] [2|3] [3|6] [6|2] [2|9]