

# Aufgabe 1 - Seilschaften

Dominic S. Meiser

## 33. Bundeswettbewerb Informatik Runde 2

1. Stein(75, Oben) fahren runter und niemand hoch.
2. Person(90, Oben) fahren runter und Stein(75, Oben) hoch.
3. Person(105, Oben) fahren runter und Person(90, Oben) hoch.
4. Stein(75, Oben) fahren runter und niemand hoch.
5. Person(195, Oben) fahren runter und Stein(75, Oben), Person(105, Oben) hoch.
6. Stein(75, Oben) fahren runter und niemand hoch.
7. Person(90, Oben) fahren runter und Stein(75, Oben) hoch.
8. Person(105, Oben) fahren runter und Person(90, Oben) hoch.
9. Stein(75, Oben) fahren runter und niemand hoch.
10. Person(90, Oben) fahren runter und Stein(75, Oben) hoch.

## Inhaltsverzeichnis

<b>1</b>	<b>Lösungsidee</b>	<b>3</b>
1.1	Lesen der Eingabedaten . . . . .	3
1.2	Breitensuche . . . . .	3
<b>2</b>	<b>Umsetzung</b>	<b>5</b>
2.1	Die Eingabedaten . . . . .	5
2.2	Die Queue . . . . .	7
2.3	Der Algorithmus . . . . .	10
<b>3</b>	<b>Beispiele</b>	<b>14</b>

## 1 Lösungsidee

In dieser Aufgabe geht es darum, Personen von einem oberen Stockwerk in ein unteres zu bringen. Dabei fahren jeweils Personen/Steine hoch und welche runter, das Gewicht oben muss größer sein als das untere, die Differenz darf jedoch den angegebenen Gewichtsunterschied  $d$  nicht unterschreiten.

Dies kann man am besten lösen, indem man einen Baum aufspannt, der jeweils die nach oben und nach unten fahrenden Personen und Steine, ich nenne diese Elemente, enthält. Bei einer Breitensuche kann man den Baum aufbauen, während man ihn durchgeht, und die erste gefundene Lösung ist automatisch die beste. Deshalb habe ich mich für eine Breitensuche entschieden.

Für die Breitensuche brauche ich zuerst eine Queue, die die gefundenen Pfade im Baum abspeichert. Die Eingabedaten haben keine Lösung, wenn die Queue leer ist und keine gefunden wurde. Am Anfang enthält die Queue genau ein Element, dass noch keinen Lösungsweg enthält und den Anfangszustand speichert. Der Zustand sollte aus Effizienzgründen in eine Liste mit den Elementen, die oben sind, und eine mit denen, die unten sind, aufgeteilt werden.

### 1.1 Lesen der Eingabedaten

Als erstes müssen die Eingabedaten eingelesen werden. Ich habe hierfür das gleiche Eingabeformat wie die Beispiele des BwInf genommen: Die erste Zeile enthält die Gewichtsschranke  $d$ , alle weiteren die Elemente, dabei steht  $P$  für Person und  $S$  für Stein, als nächstes kommt das Gewicht, und zum Schluss entweder ein  $\wedge$  wenn das Element oben ist, ansonsten ein  $\_$ . Die Leerzeichen sind unnötig. Alle Zeilen außer der ersten folgen also diesem regulären Ausdruck:

```
([PS])\s*(\d+)\s*([\w^])
```

Sollte der Benutzer die Eingabedaten von der Kommandozeile aus eingeben muss als letzte Zeile ein  $\_$  eingegeben werden. Ansonsten kann die Eingabe mit einem normalen *EOF* aufhören.

### 1.2 Breitensuche

Nachdem ich die Eingabedaten eingelesen habe, teile ich zuerst die Elemente in die ein, die sich oben befinden, und die, die sich unten befinden. Bevor ich mit der Breitensuche anfangen, berechne ich noch das Maximalgewicht eines Korbes unter der Annahme, dass Personen mitfahren. Dieses ist die Hälfte des Gewichts aller Elemente  $+ d$ .

Als erstes erstelle ich nun die Queue. Die Queue sollte einer LinkedList ähneln, allerdings noch eine ClosedList in Form eines HashSets enthalten, dass die bereits angeschauten Zwischenzustände enthält. Wenn ich jetzt ein Element aus der Queue anfordere, überprüft diese zunächst, ob das Element bereits angeschaut wurde. Ist das der Fall, wird das nächste Element angeschaut, bis ich ein Element gefunden habe, dass noch nicht in der ClosedList enthalten ist. Dieses Element wird dann in der ClosedList gespeichert und zurückgegeben. Damit habe ich eine Maximale Laufzeit  $\mathcal{O}(n)$ , wobei  $n$  die Größe der OpenList ist; die `contains()`-Funktion der ClosedList ist ja konstant. Im Normalfall ist die Laufzeit aber deutlich kleiner und fast konstant. Die Methode, die ein Element an das Ende der Queue anhängt, ist ebenfalls konstant, weil nur überprüft werden muss, ob das Element schon in der ClosedList ist.

Anschließend iteriere ich solange durch alle Elemente in der Queue, bis diese leer ist. Zunächst bilde ich alle Sublisten der Liste mit den Elementen oben und unten. Sollte jedoch irgendwo ein Stein im Aufzug sein bzw. oben/unten sein, aber es keine Person gibt, die diesen Stein ein-/ausladen könnte, muss der alte Korb wiederverwendet werden. Nun iteriere ich genestet durch diese beiden Listen und schaue mir jeweils die Kombinationen an. Allerdings kann ich eine gebildete Sublist ausschließen wenn diese mindestens eine Person enthält und größer als das berechnete Maximalgewicht eines Korbes ist. Im Falle der zweiten Schleife kann ich diese Sublist einfach löschen, dann muss ich mir diese beim nächsten Durchlauf der äußeren Schleife nicht noch einmal

anschauen. Bevor ich die Kombination als gültig erklären kann, muss ich noch schauen, dass das Gewicht der herunterfahrenden Elemente nicht zu viel und auch nicht zu wenig ist, damit der Korb fährt, mit Personen aber nicht zu schnell ist (bei Steinen ist das egal). Nun kann ich, insofern nach diesem Schritt keine Personen oben sind, diese Lösung als beste zurückgeben, andernfalls wird dieser Lösungsansatz der Queue hinzugefügt. Um meine Lösung am Schluss noch zu überprüfen, sollte das Programm bei der Ausgabe der Lösung den Lösungsweg noch einmal nachgehen und den Endzustand ausgeben.

## 2 Umsetzung

Ich habe die Lösungsidee in C++11 mit dem Framework Qt 5 umgesetzt. Die Beispieleingabedaten des BwInf habe ich mithilfe eines Qt Resource Files in das Programm eingebunden:

```
<RCC>
    <qresource prefix="/examples">
        <file>seilschaften0.txt</file>
        <file>seilschaften1.txt</file>
        <file>seilschaften2.txt</file>
        <file>seilschaften3.txt</file>
        <file>seilschaften4.txt</file>
        <file>seilschaften5.txt</file>
        <file>seilschaften6.txt</file>
    </qresource>
</RCC>
```

Anschließend kann ich die Argumente mithilfe von `QCommandLineParser` parsen:

```
QCommandLineParser parser;
parser.setApplicationDescription("Aufgabe1-33. BwInf");
parser.addHelpOption();
parser.addPositionalArgument("file",
    "Die Datei, aus der die Eingabedaten gelesen werden sollen.", "[<file>]");
QCommandLineOption verboseOption(QStringList() << "v" << "verbose",
    "Gibt genauere Details zum aktuell laufenden Vorgang aus.");
parser.addOption(verboseOption);
QCommandLineOption exampleOption(QStringList() << "e" << "example",
    "Beispiel Nummer n vom BwInf benutzen.", "n");
parser.addOption(exampleOption);
parser.process(app);
```

Nun erstelle ich eine Instanz des Algorithmus (Klasse `SeilschaftenSolver`) und der Klasse `InputDataParser`. Sollte eine Eingabedatei angegeben sein, wird der Inhalt dieser Datei eingelesen. Ist die Option `-e` angegeben, wird das angegebene Beispiel benutzt. Ansonsten wird der Benutzer interaktiv nach den Eingabedaten gefragt:

```
SeilschaftenSolver *solver = new SeilschaftenSolver(verbose);
InputDataParser data(solver, verbose);

QStringList args = parser.positionalArguments();
if (args.length() > 0)
    data.parse(args[0]);
else if (parser.isSet(exampleOption))
    data.parse(":/examples/seilschaften" + parser.value(exampleOption) + ".txt");
else
    data.query();
```

Die Klasse `InputDataParser` startet danach automatisch den Algorithmus.

### 2.1 Die Eingabedaten

Wie eben schon erwähnt, ist die Klasse `InputDataParser` für das Parsen der Eingabedaten zuständig. Dabei liest sie die Daten in die Struktur `InputData`, die wirklich nur das Nötigste speichert:

```
struct InputData
{
    unsigned int d;
    QList<Element> elements;
```

};

Die Klasse `Element` speichert Gewicht, Typ und Status eines Elements und definiert noch ein paar nützliche, selbsterklärende Funktionen:

```
class Element
{
public:
    enum Type : char
    {
        Stone = 'S',
        Person = 'P'
    };

    enum State : char
    {
        Up = '^',
        Down = '_'
    };

    Type type;
    unsigned int weight;
    State state;

    bool isUp () const { return (state == Up); }
    bool isDown () const { return (state == Down); }
    bool isStone () const { return (type == Stone); }
    bool isPerson () const { return (type == Person); }

    bool operator== (const Element &other) const
    {
        return ((type == other.type)
            && (weight == other.weight)
            && (state == other.state));
    }

    QString toString () const
    {
        return QString("%1(%2,%3)")
            .arg(
                (type == Stone ? "Stein" : type == Person ? "Person" : "Unknown"),
                QString::number(weight),
                (state == Up ? "Oben" : state == Down ? "Unten" : "Unknown")
            );
    }
};
```

Das Parsen der Eingabedaten funktioniert sehr einfach mit dem bereits in der Lösungsidee genannten regulären Ausdruck:

```
// Datei öffnen
QFile f(file);
ERROR_IF(!f.open(QIODevice::ReadOnly), "Die Datei " + file
    + " kann entweder nicht gefunden oder nicht gelesen werden.")

InputData inputdata;

// Zeile 1: Schranke für Gewichtsunterschied d
QString line = f.readLine().trimmed();
```

```

bool ok = false;
inputdata.d = line.toInt(&ok, 10);
ERROR_IF(!ok, file + ":1: Schranke kann nicht als Dezimalzahl eingelesen werden.")

// Weitere Zeilen: jeweils Angabe zu Gewicht und Ausgangsposition einer Person bzw.
// eines Steins
// NOTE: Leerzeilen werden wie Zeilenende betrachtet
QRegExp regex("([PS])\\s*(\\d+)\\s*([_~])");
for (int i = 2; !(line = f.readLine().trimmed()).isEmpty(); i++)
{
    ERROR_IF(!regex.exactMatch(line), file + ":" + i
        + ": Die Zeile passt nicht auf das vorgegebene Muster '"
        + regex.pattern() + "'.");
    inputdata.elements.append({ (regex.cap(1) == "P" ? Element::Person : Element::Stone),
        regex.cap(2).toUInt(0, 10),
        (regex.cap(3) == "_" ? Element::Down : Element::Up) });
}

// Datei schließen
printf("Eingabedaten erfolgreich aus der Datei %s gelesen.\n", file.toString().data());
f.close();

// Eingabedaten weiterleiten
_solver->execute(inputdata);

```

Das interaktive Abfragen der Eingabedaten funktioniert ähnlich, nur dass Fehler in den Zeilen toleriert werden und die Eingabe mit einem . aufhören muss.

## 2.2 Die Queue

Die Queue besteht, wie in der Lösungsidee beschrieben, aus einer OpenList und einer ClosedList. Die OpenList habe ich als LinkedList implementiert wie in der Lösungsidee vorgeschlagen. Dazu habe ich eine weitere Klasse, `FilteredQueueData`, angelegt, die einen Pointer auf das vorherige und auf das nächste Element enthält, und außerdem natürlich noch das Element in der Liste selbst speichert. Die Klasse `FilteredQueue` enthält einen Pointer auf das erste und einen auf das letzte Element in der Queue, d.h. es handelt sich hierbei streng genommen um eine Deque (double ended Queue), ich werde jedoch weiterhin Queue schreiben. Zusätzlich zu diesen beiden Pointern speichert die Queue noch die Größe der OpenList sowie die ClosedList als `std::unordered_set`.

Wenn ich nun das erste Element aus der Queue haben will, egal ob ich es nur zurückgeben oder direkt mit entfernen will, hole ich solange ein Element von vorne (und entferne es bei Bedarf), bis ich eins gefunden habe, dass noch nicht in der ClosedList abgespeichert ist. Dieses speichere ich dann in der ClosedList und gebe es zurück. Beim Hinzufügen eines Elementes, dass noch nicht in der ClosedList enthalten ist, füge ich dieses einfach als nächstes Element des letzten Elements an und speichere es als letztes Element in der Queue. Hier ist der kommentierte Quelltext der beiden Klassen:

```

#define CONTAINS(list, elem) (list.find(elem) != list.end())

/**
 * FilteredQueueData um ein Element in der FilteredQueue zu speichern.
 */
template <typename T>
class FilteredQueueData
{
public:
    FilteredQueueData (const T &value) : prev(0), next(0), data(value)
    {

```

```

    }

    /// previous ClosedListData
    FilteredQueueData *prev;
    /// next ClosedListData
    FilteredQueueData *next;
    /// Die Daten dieses Elements
    T data;
};

/**
 * FilteredQueue, die als LinkedList implementiert ist und eine ClosedList in Form eines
 * HashSet's enthält.
 */
template <typename T>
class FilteredQueue
{
public:
    /** Erstellt eine neue, leere FilteredQueue. */
    FilteredQueue() : _first(0), _last(0), _size(0)
    {
    }

    /** Löscht alle Elemente in der Liste. */
    ~FilteredQueue ()
    {
        while (_first)
        {
            FilteredQueueData<T> *old = _first;
            _first = _first->next;
            delete old;
        }
    }

    /** Gibt an, ob die Liste leer ist oder nicht. */
    bool isEmpty () const { return (size() == 0); }

    /** Gibt das erste Element zurück. */
    T first () const
    {
        // überprüfen, dass die Liste nicht leer ist
        Q_ASSERT(!isEmpty());

        // das erste Element in der Liste suchen, dass noch nicht angeschaut wurde, und
        // anschließend speichern dass das Element angeschaut wurde
        T elem;
        FilteredQueueData<T> *data;
        do
        {
            elem = data->data;
            data = data->next;
        } while (CONTAINS(closedlist, elem) && data);
        closedlist.insert(elem);
        return elem;
    }

    /** Gibt das erste Element zurück und entfernt es. */
    T pop_first ()

```



```

{
    // überprüfen, dass die Liste nicht leer ist
    Q_ASSERT(!isEmpty());

    // das erste Element in der Liste suchen, das noch nicht angeschaut wurde, und
    // anschließend speichern dass das Element angeschaut wurde. Alle bereits
    // angeschauten Elemente inklusive dem gefundenen entfernen.
    T elem;
    do
    {
        elem = _first->data;
        FilteredQueueData<T> *old = _first;
        _first = _first->next;
        delete old;
        if (_first)
            _first->prev = 0;
        _size--;
    } while (CONTAINS(closedlist, elem) && _first);
    closedlist.insert(elem);
    return elem;
}

/** Fügt das Element value ans Ende der Liste hinzu. */
void push_back (const T &value)
{
    // wenn value schon angeschaut wurde, nichts machen
    if (CONTAINS(closedlist, value))
        return;

    // wenn die Liste leer ist, _last und _first einfach mit den richtigen Daten
    // füttern
    if (!_first || !_last)
    {
        _first = new FilteredQueueData<T>(value);
        _last = _first;
    }

    // ansonsten das Element hinten an die Liste anfügen
    else
    {
        FilteredQueueData<T> *elem = new FilteredQueueData<T>(value);
        elem->prev = _last;
        _last->next = elem;
        _last = elem;
    }

    // size inkrementieren
    _size++;
}

/** Fügt das Element value ans Ende der Liste hinzu. */
FilteredQueue& operator << (const T &value) { push_back(value); return *this; }

/** Gibt die Anzahl der in der Liste gespeicherten Elemente zurück. */
int size () const { return _size; }

private:
    FilteredQueueData<T> *_first;

```

```

    FilteredQueueData<T> *_last;
    uint _size;

    std::unordered_set<T> closedlist;
};

```

## 2.3 Der Algorithmus

Nachdem die Eingabedaten gelesen wurden, ruft die Klasse InputDataParser die Methode `execute()` der übergebenen SeilschaftenSolver-Instanz mit den geparsen Eingabedaten auf. Diese sortiert die Elemente in die, die oben sind, und die, die unten sind. Danach wird noch `maxweight`, das maximale Gewicht eines Korbs, wie in der Lösungsidee beschrieben berechnet, bevor der Algorithmus aufgerufen wird:

```

d = data.d;
for (Element e : data.elements)
{
    if (e.isUp())
        up << e;
    else
        down << e;
    weight += e.weight;
}
maxweight = ceil(weight / 2.0) + d;

```

Anschließend wird die `solve()`-Methode aufgerufen, die den Algorithmus enthält. Diese erstellt zuerst eine `FilteredQueue<Solution>` und fügt dort eine leere Lösung mit dem Ausgangszustand ein. Darauf folgt eine `for`-Schleife, die so lange iteriert, bis die Queue leer ist. Als nächstes bilde ich dann die Sublisten:

```

FilteredQueue<Solution> pathes;
pathes << Solution(up, down);

// jeden gefundenen Pfad weitergehen bis eine Lösung gefunden wurde
for (uint itcount = 1; !pathes.isEmpty(); itcount++)
{
    Solution solution = pathes.pop_first();

    if (itcount > 1)
        printf("\x1b[1A");
    printf("\x1b[2KIteration: \t\t%u\t\tQueue Size: \t\t%d\n", itcount, pathes.size());

    // Sublisten für die up und down Liste finden
    QList<QList<Element>> sup = sublists(solution.up);
    QList<QList<Element>> sdown = sublists(solution.down);
}

```

Die `sublists()`-Methode fügt zunächst ein leeres Element hinzu. Danach nimmt sie jedes Element aus der ursprünglichen Liste und nimmt nun jede schon gefundene Sublist und fügt das Element hinzu, sodass eine neue Sublist entsteht, die dann wiederum an die Liste mit den Sublisten angehängt wird:

```

template <typename E>
QList<QList<E>> sublists (QList<E> list)
{
    QList<QList<E>> subsets;
    QList<E> empty;
    subsets << empty;
}

```

```

for (int i = 0; i < list.size(); i++)
{
    QList<QList<E>> subsetsTemp = subsets;
    for (int j = 0; j < subsetsTemp.size(); j++)
        subsetsTemp[j] << list[i];
    subsets << subsetsTemp;
}

return subsets;
}

```

Als nächstes überprüfe ich, ob ein Korb so bleiben muss, wie er vorher war, weil keine Person zum Ein-/Ausladen da ist. Dies passiert ganz einfach, indem ich, sollte oben oder unten keine Person sein, den alten Korb benutze:

```

bool nopersonsup = true, nopersonsdown = true;
for (Element e : solution.up)
{
    if (e.isPerson())
    {
        nopersonsup = false;
        break;
    }
}
for (Element e : solution.down)
{
    if (e.isPerson())
    {
        nopersonsdown = false;
        break;
    }
}

if (nopersonsup)
    sup = QList<QList<Element>>() << solution.isEmpty() ? QList<Element>()
                                                         : solution.last().goingup;

if (nopersonsdown)
    sdown = QList<QList<Element>>() << solution.isEmpty() ? QList<Element>()
                                                         : solution.last().goingdown;

```

Dann iteriere ich durch alle Sublisten-Kombinationen, wie auch in der Lösungsidee beschrieben, und kloniere die aktuelle Solution und verschiebe, sollte ich den Zug nicht vorher als unmöglich eingestuft haben, die Elemente an die neue Position nach dem Schritt:

```

for (QList<Element> goingdown : sup)
{
    // Überprüfen dass goingdown nicht zu schwer
    uint weightgoingdown = 0;
    bool stoneonlyup = true;
    for (Element e : goingdown)
    {
        weightgoingdown += e.weight;
        if (e.isPerson())
            stoneonlyup = false;
    }
    if (!stoneonlyup && (weightgoingdown > maxweight))
        continue;
}

```

```

// clone solution
Solution s0(solution);

// Die Elemente aus goingdown nach unten verschieben
for (Element e : goingdown)
{
    s0.up.removeOne(e);
    s0.down << e;
}

// sdown-Sublisten durchiterieren, zu schwere direkt löschen
for (int j = 0; j < sdown.size(); )
{
    QList<Element> goingup = sdown[j];
    if (goingdown.isEmpty() && goingup.isEmpty())
    {
        j++;
        continue;
    }

    // Überprüfen dass goingup nicht zu schwer bzw. d nicht überschritten ist
    uint weightdown = 0;
    bool stoneonlydown = true;
    for (Element e : goingup)
    {
        weightdown += e.weight;
        if (e.isPerson())
            stoneonlydown = false;
    }
    if ((weightgoingdown <= weightdown) // Die Körbe fahren nur wenn das Gewicht
                                        // oben größer ist
        || (!(stoneonlydown && stoneonlyup)
            && (weightgoingdown - weightdown > d))) // d überschritten
    {
        j++;
        continue;
    }
    else if (!stoneonlydown && (weightdown > maxweight))
    {
        sdown.removeAt(j);
        continue;
    }

    // clone solution
    Solution s(s0);
    s << Move{ goingdown, goingup };

    // Die Elemente von goingup nach oben verschieben
    for (Element e : goingup)
    {
        s.down.removeOne(e);
        s.up << e;
    }
}

```

Nun kann ich überprüfen, ob die Aufgabe nach diesem Schritt beendet ist. Dafür schaue ich einfach, ob oben noch Personen sind. Ist dies nicht der Fall, speichere ich die Lösung und returne. Ansonsten füge ich die Solution hinten an die Queue an:

```

// Wenn keine Personen mehr in der oberen Liste sind abbrechen

```

```

        bool end = true;
        for (Element e : s.up)
        {
            if (e.isPerson())
            {
                end = false;
                break;
            }
        }
        if (end)
        {
            this->solution = s;
            return;
        }

        // Ansonsten s zu den verfügbaren Pfaden adden
        pathes.push_back(s);

        // increase j
        j++;
    } // for j=0; j<sdown.size;
} // for goingdown : sdown
} // for solution : pathes

```

Die execute()-Methode gibt nun die Lösung, die der Algorithmus gefunden hat, aus. Ist diese leer, hat der Algorithmus keine Lösung gefunden. Dabei werden die Schritte noch einmal durchgegangen und der Endzustand ausgegeben:

```

sout << "░--░L\x6" "sung:\n";
if (!solution.isEmpty())
{
    uint count = 0;
    for (Move m : solution)
    {
        sout << "░░░" << ++count << ".\t\t" << m.toString() << "\n";
        for (Element e : m.goingup)
        {
            down.removeOne(e);
            up << e;
            e.state = Element::Up;
        }
        for (Element e : m.goingdown)
        {
            up.removeOne(e);
            down << e;
            e.state = Element::Down;
        }
    }
}
sout << "░--░Endzustand:\n";
printStats(&sout);

```

### 3 Beispiele

Hier sind einige Beispiele. Die Zustände der Elemente in der Ausgabe sind die Anfangszustände, nicht die aktuellen!

```
$ ./seilschaften -e 0
Eingabedaten erfolgreich aus der Datei ./examples/seilschaften0.txt gelesen.
-- Eingabedaten erhalten:
- Gewichtsschranke d: 15
- Status:
  OBEN:
    - Person(195, Oben)
    - Person(105, Oben)
    - Person(90, Oben)
    - Stein(75, Oben)
  UNTEN:
-- Starte Algorithmus (Breitensuche)
Iteration:      18          Queue Size:      2
-- Algorithmus beendet, time consumed: 0.002 sec
-- Lösung:
- 1.    Stein(75, Oben) fahren runter und niemand hoch.
- 2.    Person(90, Oben) fahren runter und Stein(75, Oben) hoch.
- 3.    Person(105, Oben) fahren runter und Person(90, Oben) hoch.
- 4.    Stein(75, Oben) fahren runter und niemand hoch.
- 5.    Person(195, Oben) fahren runter und Stein(75, Oben), Person(105, Oben) hoch.
- 6.    Stein(75, Oben) fahren runter und niemand hoch.
- 7.    Person(90, Oben) fahren runter und Stein(75, Oben) hoch.
- 8.    Person(105, Oben) fahren runter und Person(90, Oben) hoch.
- 9.    Stein(75, Oben) fahren runter und niemand hoch.
- 10.   Person(90, Oben) fahren runter und Stein(75, Oben) hoch.
-- Endzustand:
  OBEN:
    - Stein(75, Oben)
  UNTEN:
    - Person(195, Oben)
    - Person(105, Oben)
    - Person(90, Oben)
```

```
$ ./seilschaften -e 1
Eingabedaten erfolgreich aus der Datei ./examples/seilschaften1.txt gelesen.
-- Eingabedaten erhalten:
- Gewichtsschranke d: 1
- Status:
  OBEN:
    - Stein(1, Oben)
    - Person(2, Oben)
    - Stein(4, Oben)
    - Person(8, Oben)
    - Stein(16, Oben)
    - Person(32, Oben)
    - Stein(64, Oben)
  UNTEN:
-- Starte Algorithmus (Breitensuche)
Iteration:      1088          Queue Size:      248
-- Algorithmus beendet, time consumed: 0.162 sec
-- Lösung:
- 1.    Stein(1, Oben) fahren runter und niemand hoch.
- 2.    Person(2, Oben) fahren runter und Stein(1, Oben) hoch.
```

```

- 3.    Stein(1, Oben), Stein(4, Oben) fahren runter und niemand hoch.
- 4.    Person(8, Oben) fahren runter und Stein(1, Oben), Stein(4, Oben), Person(2, Oben) hoch.
- 5.    Stein(1, Oben) fahren runter und niemand hoch.
- 6.    Person(2, Oben) fahren runter und Stein(1, Oben) hoch.
- 7.    Stein(1, Oben), Stein(4, Oben), Stein(16, Oben) fahren runter und niemand hoch.
- 8.    Person(32, Oben) fahren runter und Stein(1, Oben), Stein(4, Oben), Stein(16, Oben), Person(2, Oben), Person(8, Oben) hoch.
- 9.    Stein(1, Oben) fahren runter und niemand hoch.
- 10.   Person(2, Oben) fahren runter und Stein(1, Oben) hoch.
- 11.   Stein(1, Oben), Stein(4, Oben) fahren runter und niemand hoch.
- 12.   Person(8, Oben) fahren runter und Stein(1, Oben), Stein(4, Oben), Person(2, Oben) hoch.
- 13.   Stein(1, Oben) fahren runter und niemand hoch.
- 14.   Person(2, Oben) fahren runter und Stein(1, Oben) hoch.
-- Endzustand:
  OBEN:
    - Stein(64, Oben)
    - Stein(16, Oben)
    - Stein(4, Oben)
    - Stein(1, Oben)
  UNTEN:
    - Person(32, Oben)
    - Person(8, Oben)
    - Person(2, Oben)

```

```

$ ./seilschaften -e 2
Eingabedaten erfolgreich aus der Datei ./examples/seilschaften2.txt gelesen.
-- Eingabedaten erhalten:
- Gewichtsschranke d: 1
- Status:
  OBEN:
    - Person(1, Oben)
    - Person(2, Oben)
    - Person(4, Oben)
    - Person(8, Oben)
    - Person(16, Oben)
    - Person(32, Oben)
    - Person(64, Oben)
    - Person(128, Oben)
    - Person(256, Oben)
  UNTEN:
-- Starte Algorithmus (Breitensuche)
Iteration:      511           Queue Size:      0
-- Algorithmus beendet, time consumed: 0.123 sec
-- Lösung:
- 1.    Person(1, Oben) fahren runter und niemand hoch.
- 2.    Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 3.    Person(1, Oben) fahren runter und niemand hoch.
- 4.    Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 5.    Person(1, Oben) fahren runter und niemand hoch.
- 6.    Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 7.    Person(1, Oben) fahren runter und niemand hoch.
- 8.    Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 9.    Person(1, Oben) fahren runter und niemand hoch.
- 10.   Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 11.   Person(1, Oben) fahren runter und niemand hoch.

```

- 12. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 13. Person(1, Oben) fahren runter und niemand hoch.
- 14. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 15. Person(1, Oben) fahren runter und niemand hoch.
- 16. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 17. Person(1, Oben) fahren runter und niemand hoch.
- 18. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 19. Person(1, Oben) fahren runter und niemand hoch.
- 20. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 21. Person(1, Oben) fahren runter und niemand hoch.
- 22. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 23. Person(1, Oben) fahren runter und niemand hoch.
- 24. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 25. Person(1, Oben) fahren runter und niemand hoch.
- 26. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 27. Person(1, Oben) fahren runter und niemand hoch.
- 28. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 29. Person(1, Oben) fahren runter und niemand hoch.
- 30. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 31. Person(1, Oben) fahren runter und niemand hoch.
- 32. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben) hoch.
- 33. Person(1, Oben) fahren runter und niemand hoch.
- 34. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 35. Person(1, Oben) fahren runter und niemand hoch.
- 36. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 37. Person(1, Oben) fahren runter und niemand hoch.
- 38. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 39. Person(1, Oben) fahren runter und niemand hoch.
- 40. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 41. Person(1, Oben) fahren runter und niemand hoch.
- 42. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 43. Person(1, Oben) fahren runter und niemand hoch.
- 44. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 45. Person(1, Oben) fahren runter und niemand hoch.
- 46. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 47. Person(1, Oben) fahren runter und niemand hoch.
- 48. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 49. Person(1, Oben) fahren runter und niemand hoch.
- 50. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 51. Person(1, Oben) fahren runter und niemand hoch.
- 52. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 53. Person(1, Oben) fahren runter und niemand hoch.
- 54. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 55. Person(1, Oben) fahren runter und niemand hoch.
- 56. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 57. Person(1, Oben) fahren runter und niemand hoch.
- 58. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 59. Person(1, Oben) fahren runter und niemand hoch.
- 60. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 61. Person(1, Oben) fahren runter und niemand hoch.
- 62. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 63. Person(1, Oben) fahren runter und niemand hoch.



- 64. Person(64, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben) hoch.
- 65. Person(1, Oben) fahren runter und niemand hoch.
- 66. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 67. Person(1, Oben) fahren runter und niemand hoch.
- 68. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 69. Person(1, Oben) fahren runter und niemand hoch.
- 70. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 71. Person(1, Oben) fahren runter und niemand hoch.
- 72. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 73. Person(1, Oben) fahren runter und niemand hoch.
- 74. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 75. Person(1, Oben) fahren runter und niemand hoch.
- 76. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 77. Person(1, Oben) fahren runter und niemand hoch.
- 78. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 79. Person(1, Oben) fahren runter und niemand hoch.
- 80. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 81. Person(1, Oben) fahren runter und niemand hoch.
- 82. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 83. Person(1, Oben) fahren runter und niemand hoch.
- 84. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 85. Person(1, Oben) fahren runter und niemand hoch.
- 86. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 87. Person(1, Oben) fahren runter und niemand hoch.
- 88. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 89. Person(1, Oben) fahren runter und niemand hoch.
- 90. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 91. Person(1, Oben) fahren runter und niemand hoch.
- 92. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 93. Person(1, Oben) fahren runter und niemand hoch.
- 94. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 95. Person(1, Oben) fahren runter und niemand hoch.
- 96. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben) hoch.
- 97. Person(1, Oben) fahren runter und niemand hoch.
- 98. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 99. Person(1, Oben) fahren runter und niemand hoch.
- 100. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 101. Person(1, Oben) fahren runter und niemand hoch.
- 102. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 103. Person(1, Oben) fahren runter und niemand hoch.
- 104. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 105. Person(1, Oben) fahren runter und niemand hoch.
- 106. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 107. Person(1, Oben) fahren runter und niemand hoch.
- 108. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 109. Person(1, Oben) fahren runter und niemand hoch.
- 110. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 111. Person(1, Oben) fahren runter und niemand hoch.
- 112. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 113. Person(1, Oben) fahren runter und niemand hoch.
- 114. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 115. Person(1, Oben) fahren runter und niemand hoch.
- 116. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 117. Person(1, Oben) fahren runter und niemand hoch.
- 118. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 119. Person(1, Oben) fahren runter und niemand hoch.
- 120. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 121. Person(1, Oben) fahren runter und niemand hoch.
- 122. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 123. Person(1, Oben) fahren runter und niemand hoch.
- 124. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 125. Person(1, Oben) fahren runter und niemand hoch.
- 126. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 127. Person(1, Oben) fahren runter und niemand hoch.
- 128. Person(128, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben), Person(64, Oben) hoch.
- 129. Person(1, Oben) fahren runter und niemand hoch.
- 130. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 131. Person(1, Oben) fahren runter und niemand hoch.
- 132. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 133. Person(1, Oben) fahren runter und niemand hoch.
- 134. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 135. Person(1, Oben) fahren runter und niemand hoch.
- 136. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 137. Person(1, Oben) fahren runter und niemand hoch.
- 138. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 139. Person(1, Oben) fahren runter und niemand hoch.
- 140. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 141. Person(1, Oben) fahren runter und niemand hoch.
- 142. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 143. Person(1, Oben) fahren runter und niemand hoch.
- 144. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 145. Person(1, Oben) fahren runter und niemand hoch.
- 146. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 147. Person(1, Oben) fahren runter und niemand hoch.
- 148. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 149. Person(1, Oben) fahren runter und niemand hoch.
- 150. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 151. Person(1, Oben) fahren runter und niemand hoch.
- 152. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 153. Person(1, Oben) fahren runter und niemand hoch.
- 154. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 155. Person(1, Oben) fahren runter und niemand hoch.
- 156. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 157. Person(1, Oben) fahren runter und niemand hoch.
- 158. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 159. Person(1, Oben) fahren runter und niemand hoch.
- 160. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben) hoch.
- 161. Person(1, Oben) fahren runter und niemand hoch.
- 162. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 163. Person(1, Oben) fahren runter und niemand hoch.
- 164. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 165. Person(1, Oben) fahren runter und niemand hoch.
- 166. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 167. Person(1, Oben) fahren runter und niemand hoch.
- 168. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0  
ben) hoch.
- 169. Person(1, Oben) fahren runter und niemand hoch.
- 170. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 171. Person(1, Oben) fahren runter und niemand hoch.
- 172. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 173. Person(1, Oben) fahren runter und niemand hoch.
- 174. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 175. Person(1, Oben) fahren runter und niemand hoch.
- 176. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,  
Oben), Person(8, Oben) hoch.
- 177. Person(1, Oben) fahren runter und niemand hoch.
- 178. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 179. Person(1, Oben) fahren runter und niemand hoch.
- 180. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 181. Person(1, Oben) fahren runter und niemand hoch.
- 182. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 183. Person(1, Oben) fahren runter und niemand hoch.
- 184. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0  
ben) hoch.
- 185. Person(1, Oben) fahren runter und niemand hoch.
- 186. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 187. Person(1, Oben) fahren runter und niemand hoch.
- 188. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 189. Person(1, Oben) fahren runter und niemand hoch.
- 190. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 191. Person(1, Oben) fahren runter und niemand hoch.
- 192. Person(64, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,  
Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben) hoch.
- 193. Person(1, Oben) fahren runter und niemand hoch.
- 194. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 195. Person(1, Oben) fahren runter und niemand hoch.
- 196. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 197. Person(1, Oben) fahren runter und niemand hoch.
- 198. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 199. Person(1, Oben) fahren runter und niemand hoch.
- 200. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0  
ben) hoch.
- 201. Person(1, Oben) fahren runter und niemand hoch.
- 202. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 203. Person(1, Oben) fahren runter und niemand hoch.
- 204. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 205. Person(1, Oben) fahren runter und niemand hoch.
- 206. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 207. Person(1, Oben) fahren runter und niemand hoch.
- 208. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,  
Oben), Person(8, Oben) hoch.
- 209. Person(1, Oben) fahren runter und niemand hoch.
- 210. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 211. Person(1, Oben) fahren runter und niemand hoch.
- 212. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 213. Person(1, Oben) fahren runter und niemand hoch.
- 214. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 215. Person(1, Oben) fahren runter und niemand hoch.
- 216. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0  
ben) hoch.
- 217. Person(1, Oben) fahren runter und niemand hoch.

- 218. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 219. Person(1, Oben) fahren runter und niemand hoch.
- 220. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 221. Person(1, Oben) fahren runter und niemand hoch.
- 222. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 223. Person(1, Oben) fahren runter und niemand hoch.
- 224. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben) hoch.
- 225. Person(1, Oben) fahren runter und niemand hoch.
- 226. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 227. Person(1, Oben) fahren runter und niemand hoch.
- 228. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 229. Person(1, Oben) fahren runter und niemand hoch.
- 230. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 231. Person(1, Oben) fahren runter und niemand hoch.
- 232. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 233. Person(1, Oben) fahren runter und niemand hoch.
- 234. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 235. Person(1, Oben) fahren runter und niemand hoch.
- 236. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 237. Person(1, Oben) fahren runter und niemand hoch.
- 238. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 239. Person(1, Oben) fahren runter und niemand hoch.
- 240. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 241. Person(1, Oben) fahren runter und niemand hoch.
- 242. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 243. Person(1, Oben) fahren runter und niemand hoch.
- 244. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 245. Person(1, Oben) fahren runter und niemand hoch.
- 246. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 247. Person(1, Oben) fahren runter und niemand hoch.
- 248. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 249. Person(1, Oben) fahren runter und niemand hoch.
- 250. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 251. Person(1, Oben) fahren runter und niemand hoch.
- 252. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 253. Person(1, Oben) fahren runter und niemand hoch.
- 254. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 255. Person(1, Oben) fahren runter und niemand hoch.
- 256. Person(256, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben), Person(64, Oben), Person(128, Oben) hoch.
- 257. Person(1, Oben) fahren runter und niemand hoch.
- 258. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 259. Person(1, Oben) fahren runter und niemand hoch.
- 260. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 261. Person(1, Oben) fahren runter und niemand hoch.
- 262. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 263. Person(1, Oben) fahren runter und niemand hoch.
- 264. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 265. Person(1, Oben) fahren runter und niemand hoch.
- 266. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 267. Person(1, Oben) fahren runter und niemand hoch.
- 268. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.

- 269. Person(1, Oben) fahren runter und niemand hoch.
- 270. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 271. Person(1, Oben) fahren runter und niemand hoch.
- 272. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 273. Person(1, Oben) fahren runter und niemand hoch.
- 274. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 275. Person(1, Oben) fahren runter und niemand hoch.
- 276. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 277. Person(1, Oben) fahren runter und niemand hoch.
- 278. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 279. Person(1, Oben) fahren runter und niemand hoch.
- 280. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0 ben) hoch.
- 281. Person(1, Oben) fahren runter und niemand hoch.
- 282. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 283. Person(1, Oben) fahren runter und niemand hoch.
- 284. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 285. Person(1, Oben) fahren runter und niemand hoch.
- 286. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 287. Person(1, Oben) fahren runter und niemand hoch.
- 288. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben) hoch.
- 289. Person(1, Oben) fahren runter und niemand hoch.
- 290. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 291. Person(1, Oben) fahren runter und niemand hoch.
- 292. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 293. Person(1, Oben) fahren runter und niemand hoch.
- 294. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 295. Person(1, Oben) fahren runter und niemand hoch.
- 296. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0 ben) hoch.
- 297. Person(1, Oben) fahren runter und niemand hoch.
- 298. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 299. Person(1, Oben) fahren runter und niemand hoch.
- 300. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 301. Person(1, Oben) fahren runter und niemand hoch.
- 302. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 303. Person(1, Oben) fahren runter und niemand hoch.
- 304. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 305. Person(1, Oben) fahren runter und niemand hoch.
- 306. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 307. Person(1, Oben) fahren runter und niemand hoch.
- 308. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 309. Person(1, Oben) fahren runter und niemand hoch.
- 310. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 311. Person(1, Oben) fahren runter und niemand hoch.
- 312. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0 ben) hoch.
- 313. Person(1, Oben) fahren runter und niemand hoch.
- 314. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 315. Person(1, Oben) fahren runter und niemand hoch.
- 316. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 317. Person(1, Oben) fahren runter und niemand hoch.
- 318. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 319. Person(1, Oben) fahren runter und niemand hoch.
- 320. Person(64, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,

Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben) hoch.

- 321. Person(1, Oben) fahren runter und niemand hoch.
- 322. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 323. Person(1, Oben) fahren runter und niemand hoch.
- 324. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 325. Person(1, Oben) fahren runter und niemand hoch.
- 326. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 327. Person(1, Oben) fahren runter und niemand hoch.
- 328. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 329. Person(1, Oben) fahren runter und niemand hoch.
- 330. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 331. Person(1, Oben) fahren runter und niemand hoch.
- 332. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 333. Person(1, Oben) fahren runter und niemand hoch.
- 334. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 335. Person(1, Oben) fahren runter und niemand hoch.
- 336. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 337. Person(1, Oben) fahren runter und niemand hoch.
- 338. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 339. Person(1, Oben) fahren runter und niemand hoch.
- 340. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 341. Person(1, Oben) fahren runter und niemand hoch.
- 342. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 343. Person(1, Oben) fahren runter und niemand hoch.
- 344. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 345. Person(1, Oben) fahren runter und niemand hoch.
- 346. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 347. Person(1, Oben) fahren runter und niemand hoch.
- 348. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 349. Person(1, Oben) fahren runter und niemand hoch.
- 350. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 351. Person(1, Oben) fahren runter und niemand hoch.
- 352. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben) hoch.
- 353. Person(1, Oben) fahren runter und niemand hoch.
- 354. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 355. Person(1, Oben) fahren runter und niemand hoch.
- 356. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 357. Person(1, Oben) fahren runter und niemand hoch.
- 358. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 359. Person(1, Oben) fahren runter und niemand hoch.
- 360. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 361. Person(1, Oben) fahren runter und niemand hoch.
- 362. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 363. Person(1, Oben) fahren runter und niemand hoch.
- 364. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 365. Person(1, Oben) fahren runter und niemand hoch.
- 366. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 367. Person(1, Oben) fahren runter und niemand hoch.
- 368. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 369. Person(1, Oben) fahren runter und niemand hoch.
- 370. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 371. Person(1, Oben) fahren runter und niemand hoch.

- 372. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 373. Person(1, Oben) fahren runter und niemand hoch.
- 374. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 375. Person(1, Oben) fahren runter und niemand hoch.
- 376. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0  
ben) hoch.
- 377. Person(1, Oben) fahren runter und niemand hoch.
- 378. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 379. Person(1, Oben) fahren runter und niemand hoch.
- 380. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 381. Person(1, Oben) fahren runter und niemand hoch.
- 382. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 383. Person(1, Oben) fahren runter und niemand hoch.
- 384. Person(128, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,  
Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben), Person(64, Oben) hoch.
- 385. Person(1, Oben) fahren runter und niemand hoch.
- 386. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 387. Person(1, Oben) fahren runter und niemand hoch.
- 388. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 389. Person(1, Oben) fahren runter und niemand hoch.
- 390. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 391. Person(1, Oben) fahren runter und niemand hoch.
- 392. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0  
ben) hoch.
- 393. Person(1, Oben) fahren runter und niemand hoch.
- 394. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 395. Person(1, Oben) fahren runter und niemand hoch.
- 396. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 397. Person(1, Oben) fahren runter und niemand hoch.
- 398. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 399. Person(1, Oben) fahren runter und niemand hoch.
- 400. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,  
Oben), Person(8, Oben) hoch.
- 401. Person(1, Oben) fahren runter und niemand hoch.
- 402. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 403. Person(1, Oben) fahren runter und niemand hoch.
- 404. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 405. Person(1, Oben) fahren runter und niemand hoch.
- 406. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 407. Person(1, Oben) fahren runter und niemand hoch.
- 408. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0  
ben) hoch.
- 409. Person(1, Oben) fahren runter und niemand hoch.
- 410. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 411. Person(1, Oben) fahren runter und niemand hoch.
- 412. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 413. Person(1, Oben) fahren runter und niemand hoch.
- 414. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 415. Person(1, Oben) fahren runter und niemand hoch.
- 416. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,  
Oben), Person(8, Oben), Person(16, Oben) hoch.
- 417. Person(1, Oben) fahren runter und niemand hoch.
- 418. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 419. Person(1, Oben) fahren runter und niemand hoch.
- 420. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 421. Person(1, Oben) fahren runter und niemand hoch.
- 422. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 423. Person(1, Oben) fahren runter und niemand hoch.

- 424. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.

- 425. Person(1, Oben) fahren runter und niemand hoch.

- 426. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 427. Person(1, Oben) fahren runter und niemand hoch.

- 428. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.

- 429. Person(1, Oben) fahren runter und niemand hoch.

- 430. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 431. Person(1, Oben) fahren runter und niemand hoch.

- 432. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.

- 433. Person(1, Oben) fahren runter und niemand hoch.

- 434. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 435. Person(1, Oben) fahren runter und niemand hoch.

- 436. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.

- 437. Person(1, Oben) fahren runter und niemand hoch.

- 438. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 439. Person(1, Oben) fahren runter und niemand hoch.

- 440. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.

- 441. Person(1, Oben) fahren runter und niemand hoch.

- 442. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 443. Person(1, Oben) fahren runter und niemand hoch.

- 444. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.

- 445. Person(1, Oben) fahren runter und niemand hoch.

- 446. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 447. Person(1, Oben) fahren runter und niemand hoch.

- 448. Person(64, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben) hoch.

- 449. Person(1, Oben) fahren runter und niemand hoch.

- 450. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 451. Person(1, Oben) fahren runter und niemand hoch.

- 452. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.

- 453. Person(1, Oben) fahren runter und niemand hoch.

- 454. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 455. Person(1, Oben) fahren runter und niemand hoch.

- 456. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.

- 457. Person(1, Oben) fahren runter und niemand hoch.

- 458. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 459. Person(1, Oben) fahren runter und niemand hoch.

- 460. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.

- 461. Person(1, Oben) fahren runter und niemand hoch.

- 462. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 463. Person(1, Oben) fahren runter und niemand hoch.

- 464. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.

- 465. Person(1, Oben) fahren runter und niemand hoch.

- 466. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 467. Person(1, Oben) fahren runter und niemand hoch.

- 468. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.

- 469. Person(1, Oben) fahren runter und niemand hoch.

- 470. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

- 471. Person(1, Oben) fahren runter und niemand hoch.

- 472. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.

- 473. Person(1, Oben) fahren runter und niemand hoch.

- 474. Person(2, Oben) fahren runter und Person(1, Oben) hoch.



```

- 475. Person(1, Oben) fahren runter und niemand hoch.
- 476. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 477. Person(1, Oben) fahren runter und niemand hoch.
- 478. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 479. Person(1, Oben) fahren runter und niemand hoch.
- 480. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,
Oben), Person(8, Oben), Person(16, Oben) hoch.
- 481. Person(1, Oben) fahren runter und niemand hoch.
- 482. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 483. Person(1, Oben) fahren runter und niemand hoch.
- 484. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 485. Person(1, Oben) fahren runter und niemand hoch.
- 486. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 487. Person(1, Oben) fahren runter und niemand hoch.
- 488. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0
ben) hoch.
- 489. Person(1, Oben) fahren runter und niemand hoch.
- 490. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 491. Person(1, Oben) fahren runter und niemand hoch.
- 492. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 493. Person(1, Oben) fahren runter und niemand hoch.
- 494. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 495. Person(1, Oben) fahren runter und niemand hoch.
- 496. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,
Oben), Person(8, Oben) hoch.
- 497. Person(1, Oben) fahren runter und niemand hoch.
- 498. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 499. Person(1, Oben) fahren runter und niemand hoch.
- 500. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 501. Person(1, Oben) fahren runter und niemand hoch.
- 502. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 503. Person(1, Oben) fahren runter und niemand hoch.
- 504. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0
ben) hoch.
- 505. Person(1, Oben) fahren runter und niemand hoch.
- 506. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 507. Person(1, Oben) fahren runter und niemand hoch.
- 508. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 509. Person(1, Oben) fahren runter und niemand hoch.
- 510. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 511. Person(1, Oben) fahren runter und niemand hoch.
-- Endzustand:
  OBEN:
  UNTEN:
    - Person(256, Oben)
    - Person(128, Oben)
    - Person(64, Oben)
    - Person(32, Oben)
    - Person(16, Oben)
    - Person(8, Oben)
    - Person(4, Oben)
    - Person(2, Oben)
    - Person(1, Oben)

```

```
$ ./seilschaften
```

```
Schranke für den Gewichtsunterschied (d): 1
```

```
Personen und Steine (Format: '([PS])s*(\d+)\s*([_~])'). Beenden mit einer Zeile die nur
einen Punkt enthält.
```

```

Seilschaften:0 > P1^
Seilschaften:1 > P2^
Seilschaften:2 > P4^
Seilschaften:3 > P8^
Seilschaften:4 > P16^
Seilschaften:5 > P32^
Seilschaften:6 > S64^
Seilschaften:7 > P128^
Seilschaften:8 > P256^
Seilschaften:9 > .
-- Eingabedaten erhalten:
- Gewichtsschranke d: 1
- Status:
  OBEN:
    - Person(1, Oben)
    - Person(2, Oben)
    - Person(4, Oben)
    - Person(8, Oben)
    - Person(16, Oben)
    - Person(32, Oben)
    - Stein(64, Oben)
    - Person(128, Oben)
    - Person(256, Oben)
  UNTEN:
-- Starte Algorithmus (Breitensuche)
Iteration:      1455      Queue Size:      6
-- Algorithmus beendet, time consumed: 0.296 sec
-- Lösung:
- 1.    Person(1, Oben) fahren runter und niemand hoch.
- 2.    Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 3.    Stein(64, Oben) fahren runter und niemand hoch.
- 4.    Person(1, Oben) fahren runter und niemand hoch.
- 5.    Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 6.    Person(1, Oben) fahren runter und niemand hoch.
- 7.    Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 8.    Person(1, Oben) fahren runter und niemand hoch.
- 9.    Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0
ben) hoch.
- 10.   Person(1, Oben) fahren runter und niemand hoch.
- 11.   Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 12.   Person(1, Oben) fahren runter und niemand hoch.
- 13.   Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 14.   Person(1, Oben) fahren runter und niemand hoch.
- 15.   Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 16.   Person(1, Oben) fahren runter und niemand hoch.
- 17.   Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,
Oben), Person(8, Oben) hoch.
- 18.   Person(1, Oben) fahren runter und niemand hoch.
- 19.   Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 20.   Person(1, Oben) fahren runter und niemand hoch.
- 21.   Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 22.   Person(1, Oben) fahren runter und niemand hoch.
- 23.   Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 24.   Person(1, Oben) fahren runter und niemand hoch.
- 25.   Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0
ben) hoch.
- 26.   Person(1, Oben) fahren runter und niemand hoch.
- 27.   Person(2, Oben) fahren runter und Person(1, Oben) hoch.

```

- 28. Person(1, Oben) fahren runter und niemand hoch.
- 29. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 30. Person(1, Oben) fahren runter und niemand hoch.
- 31. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 32. Person(1, Oben) fahren runter und niemand hoch.
- 33. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben) hoch.
- 34. Person(1, Oben) fahren runter und niemand hoch.
- 35. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 36. Person(1, Oben) fahren runter und niemand hoch.
- 37. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 38. Person(1, Oben) fahren runter und niemand hoch.
- 39. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 40. Person(1, Oben) fahren runter und niemand hoch.
- 41. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 42. Person(1, Oben) fahren runter und niemand hoch.
- 43. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 44. Person(1, Oben) fahren runter und niemand hoch.
- 45. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 46. Person(1, Oben) fahren runter und niemand hoch.
- 47. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 48. Person(1, Oben) fahren runter und niemand hoch.
- 49. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 50. Person(1, Oben) fahren runter und niemand hoch.
- 51. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 52. Person(1, Oben) fahren runter und niemand hoch.
- 53. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 54. Person(1, Oben) fahren runter und niemand hoch.
- 55. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 56. Person(1, Oben) fahren runter und niemand hoch.
- 57. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 58. Person(1, Oben) fahren runter und niemand hoch.
- 59. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 60. Person(1, Oben) fahren runter und niemand hoch.
- 61. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 62. Person(1, Oben) fahren runter und niemand hoch.
- 63. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 64. Person(1, Oben) fahren runter und niemand hoch.
- 65. Person(128, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben), Stein(64, Oben) hoch.
- 66. Stein(64, Oben) fahren runter und niemand hoch.
- 67. Person(1, Oben) fahren runter und niemand hoch.
- 68. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 69. Person(1, Oben) fahren runter und niemand hoch.
- 70. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 71. Person(1, Oben) fahren runter und niemand hoch.
- 72. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 73. Person(1, Oben) fahren runter und niemand hoch.
- 74. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 75. Person(1, Oben) fahren runter und niemand hoch.
- 76. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 77. Person(1, Oben) fahren runter und niemand hoch.
- 78. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 79. Person(1, Oben) fahren runter und niemand hoch.

- 80. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 81. Person(1, Oben) fahren runter und niemand hoch.
- 82. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 83. Person(1, Oben) fahren runter und niemand hoch.
- 84. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 85. Person(1, Oben) fahren runter und niemand hoch.
- 86. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 87. Person(1, Oben) fahren runter und niemand hoch.
- 88. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 89. Person(1, Oben) fahren runter und niemand hoch.
- 90. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 91. Person(1, Oben) fahren runter und niemand hoch.
- 92. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 93. Person(1, Oben) fahren runter und niemand hoch.
- 94. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 95. Person(1, Oben) fahren runter und niemand hoch.
- 96. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 97. Person(1, Oben) fahren runter und niemand hoch.
- 98. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben) hoch.
- 99. Person(1, Oben) fahren runter und niemand hoch.
- 100. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 101. Person(1, Oben) fahren runter und niemand hoch.
- 102. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 103. Person(1, Oben) fahren runter und niemand hoch.
- 104. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 105. Person(1, Oben) fahren runter und niemand hoch.
- 106. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 107. Person(1, Oben) fahren runter und niemand hoch.
- 108. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 109. Person(1, Oben) fahren runter und niemand hoch.
- 110. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 111. Person(1, Oben) fahren runter und niemand hoch.
- 112. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 113. Person(1, Oben) fahren runter und niemand hoch.
- 114. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 115. Person(1, Oben) fahren runter und niemand hoch.
- 116. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 117. Person(1, Oben) fahren runter und niemand hoch.
- 118. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 119. Person(1, Oben) fahren runter und niemand hoch.
- 120. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 121. Person(1, Oben) fahren runter und niemand hoch.
- 122. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 123. Person(1, Oben) fahren runter und niemand hoch.
- 124. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 125. Person(1, Oben) fahren runter und niemand hoch.
- 126. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 127. Person(1, Oben) fahren runter und niemand hoch.
- 128. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 129. Person(1, Oben) fahren runter und niemand hoch.
- 130. Person(256, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben), Stein(64, Oben), Person(128, Oben) hoch.

```

Oben) hoch.
- 131. Stein(64, Oben) fahren runter und niemand hoch.
- 132. Person(1, Oben) fahren runter und niemand hoch.
- 133. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 134. Person(1, Oben) fahren runter und niemand hoch.
- 135. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 136. Person(1, Oben) fahren runter und niemand hoch.
- 137. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 138. Person(1, Oben) fahren runter und niemand hoch.
- 139. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0
ben) hoch.
- 140. Person(1, Oben) fahren runter und niemand hoch.
- 141. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 142. Person(1, Oben) fahren runter und niemand hoch.
- 143. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 144. Person(1, Oben) fahren runter und niemand hoch.
- 145. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 146. Person(1, Oben) fahren runter und niemand hoch.
- 147. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,
Oben), Person(8, Oben) hoch.
- 148. Person(1, Oben) fahren runter und niemand hoch.
- 149. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 150. Person(1, Oben) fahren runter und niemand hoch.
- 151. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 152. Person(1, Oben) fahren runter und niemand hoch.
- 153. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 154. Person(1, Oben) fahren runter und niemand hoch.
- 155. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0
ben) hoch.
- 156. Person(1, Oben) fahren runter und niemand hoch.
- 157. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 158. Person(1, Oben) fahren runter und niemand hoch.
- 159. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 160. Person(1, Oben) fahren runter und niemand hoch.
- 161. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 162. Person(1, Oben) fahren runter und niemand hoch.
- 163. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,
Oben), Person(8, Oben), Person(16, Oben) hoch.
- 164. Person(1, Oben) fahren runter und niemand hoch.
- 165. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 166. Person(1, Oben) fahren runter und niemand hoch.
- 167. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 168. Person(1, Oben) fahren runter und niemand hoch.
- 169. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 170. Person(1, Oben) fahren runter und niemand hoch.
- 171. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0
ben) hoch.
- 172. Person(1, Oben) fahren runter und niemand hoch.
- 173. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 174. Person(1, Oben) fahren runter und niemand hoch.
- 175. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 176. Person(1, Oben) fahren runter und niemand hoch.
- 177. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 178. Person(1, Oben) fahren runter und niemand hoch.
- 179. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,
Oben), Person(8, Oben) hoch.
- 180. Person(1, Oben) fahren runter und niemand hoch.
- 181. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

```

- 182. Person(1, Oben) fahren runter und niemand hoch.
- 183. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 184. Person(1, Oben) fahren runter und niemand hoch.
- 185. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 186. Person(1, Oben) fahren runter und niemand hoch.
- 187. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 188. Person(1, Oben) fahren runter und niemand hoch.
- 189. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 190. Person(1, Oben) fahren runter und niemand hoch.
- 191. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 192. Person(1, Oben) fahren runter und niemand hoch.
- 193. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 194. Person(1, Oben) fahren runter und niemand hoch.
- 195. Person(128, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben), Person(32, Oben), Stein(64, Oben) hoch.
- 196. Person(1, Oben) fahren runter und niemand hoch.
- 197. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 198. Person(1, Oben) fahren runter und niemand hoch.
- 199. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 200. Person(1, Oben) fahren runter und niemand hoch.
- 201. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 202. Person(1, Oben) fahren runter und niemand hoch.
- 203. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 204. Person(1, Oben) fahren runter und niemand hoch.
- 205. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 206. Person(1, Oben) fahren runter und niemand hoch.
- 207. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 208. Person(1, Oben) fahren runter und niemand hoch.
- 209. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 210. Person(1, Oben) fahren runter und niemand hoch.
- 211. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben) hoch.
- 212. Person(1, Oben) fahren runter und niemand hoch.
- 213. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 214. Person(1, Oben) fahren runter und niemand hoch.
- 215. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 216. Person(1, Oben) fahren runter und niemand hoch.
- 217. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 218. Person(1, Oben) fahren runter und niemand hoch.
- 219. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben) hoch.
- 220. Person(1, Oben) fahren runter und niemand hoch.
- 221. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 222. Person(1, Oben) fahren runter und niemand hoch.
- 223. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 224. Person(1, Oben) fahren runter und niemand hoch.
- 225. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 226. Person(1, Oben) fahren runter und niemand hoch.
- 227. Person(32, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, Oben), Person(8, Oben), Person(16, Oben) hoch.
- 228. Person(1, Oben) fahren runter und niemand hoch.
- 229. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 230. Person(1, Oben) fahren runter und niemand hoch.
- 231. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 232. Person(1, Oben) fahren runter und niemand hoch.
- 233. Person(2, Oben) fahren runter und Person(1, Oben) hoch.

```

- 234. Person(1, Oben) fahren runter und niemand hoch.
- 235. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0
ben) hoch.
- 236. Person(1, Oben) fahren runter und niemand hoch.
- 237. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 238. Person(1, Oben) fahren runter und niemand hoch.
- 239. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 240. Person(1, Oben) fahren runter und niemand hoch.
- 241. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 242. Person(1, Oben) fahren runter und niemand hoch.
- 243. Person(16, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4,
Oben), Person(8, Oben) hoch.
- 244. Person(1, Oben) fahren runter und niemand hoch.
- 245. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 246. Person(1, Oben) fahren runter und niemand hoch.
- 247. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 248. Person(1, Oben) fahren runter und niemand hoch.
- 249. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 250. Person(1, Oben) fahren runter und niemand hoch.
- 251. Person(8, Oben) fahren runter und Person(1, Oben), Person(2, Oben), Person(4, 0
ben) hoch.
- 252. Person(1, Oben) fahren runter und niemand hoch.
- 253. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 254. Person(1, Oben) fahren runter und niemand hoch.
- 255. Person(4, Oben) fahren runter und Person(1, Oben), Person(2, Oben) hoch.
- 256. Person(1, Oben) fahren runter und niemand hoch.
- 257. Person(2, Oben) fahren runter und Person(1, Oben) hoch.
- 258. Person(1, Oben) fahren runter und niemand hoch.
-- Endzustand:
  OBEN:
    - Stein(64, Oben)
  UNTEN:
    - Person(256, Oben)
    - Person(128, Oben)
    - Person(32, Oben)
    - Person(16, Oben)
    - Person(8, Oben)
    - Person(4, Oben)
    - Person(2, Oben)
    - Person(1, Oben)

```

```

$ ./seilschaften -e 5
Eingabedaten erfolgreich aus der Datei :/examples/seilschaften5.txt gelesen.
-- Eingabedaten erhalten:
- Gewichtsschranke d: 20
- Status:
  OBEN:
    - Person(109, Oben)
    - Person(156, Oben)
    - Person(55, Oben)
    - Person(149, Oben)
    - Person(185, Oben)
  UNTEN:
    - Stein(120, Unten)
    - Person(98, Unten)
-- Starte Algorithmus (Breitensuche)
Iteration:      4807      Queue Size:      0
-- Algorithmus beendet, time consumed: 0.962 sec

```

```
-- Lösung:
-- Endzustand:
  OBEN:
    - Person(109, Oben)
    - Person(156, Oben)
    - Person(55, Oben)
    - Person(149, Oben)
    - Person(185, Oben)
  UNTEN:
    - Stein(120, Unten)
    - Person(98, Unten)
```

```
$ ./seilschaften -e 6
Eingabedaten erfolgreich aus der Datei ./examples/seilschaften6.txt gelesen.
-- Eingabedaten erhalten:
  - Gewichtsschranke d: 20
  - Status:
    OBEN:
      - Person(109, Oben)
      - Person(156, Oben)
      - Person(55, Oben)
      - Person(149, Oben)
      - Person(185, Oben)
    UNTEN:
      - Stein(120, Unten)
      - Person(85, Unten)
-- Starte Algorithmus (Breitensuche)
Iteration:      1905      Queue Size:      3918
-- Algorithmus beendet, time consumed: 0.268 sec
-- Lösung:
  - 1.    Person(55, Oben), Person(156, Oben) fahren runter und Person(85, Unten), Stein(
120, Unten) hoch.
  - 2.    Stein(120, Unten) fahren runter und niemand hoch.
  - 3.    Person(185, Oben) fahren runter und Stein(120, Unten), Person(55, Oben) hoch.
  - 4.    Stein(120, Unten) fahren runter und niemand hoch.
  - 5.    Person(55, Oben), Person(149, Oben), Person(109, Oben) fahren runter und Stein(
120, Unten), Person(185, Oben) hoch.
  - 6.    Stein(120, Unten) fahren runter und niemand hoch.
  - 7.    Person(185, Oben) fahren runter und Stein(120, Unten), Person(55, Oben) hoch.
  - 8.    Stein(120, Unten) fahren runter und niemand hoch.
  - 9.    Person(55, Oben), Person(85, Unten) fahren runter und Stein(120, Unten) hoch.
-- Endzustand:
  OBEN:
    - Stein(120, Unten)
  UNTEN:
    - Person(156, Oben)
    - Person(109, Oben)
    - Person(149, Oben)
    - Person(185, Oben)
    - Person(85, Unten)
    - Person(55, Oben)
```