

Nelze

Created	@Nov 28, 2019 4:06 PM
Tags	Homework

Zadanie

Máme 4 krabičky s číslami (2, 4, 5, 7) a 4 krabičky s operátormi (+, +, *, *). Našou úlohou je nájsť všetky čísla, ku ktorým sa nemôžeme dostať operáciami na týchto číslach a na novovzniknutých číslach.

Graf stavov

Moje riešenie spočíva v zostavení úplného grafu všetkých možných stavov tejto hry. Každý stav sa dá popísať dvoma vlastnosťami:

- **numbers** — čísla, s ktorými v tomto stave pracujem
- **operators** — operátori, s ktorými v tomto stave pracujem

Každý stav má unikátnu kombináciu oboch vlastností.

Následne som si vytvoril množinu **A** všetkých čísel vo všetkých stavoch. Táto množina obsahuje všetky čísla, ktoré sa dajú pomocou daných čísel a operátorov zostaviť.

Stačí už len vypočítať: `<200; 300> \ A`

JavaScript Code

```
const unique = arr => Array.from(new Set(arr));

const removeOne = (arr, el) => {
  let removed = false;
  const res = [];

  for (let i = 0; i < arr.length; i++) {
    if (!removed && arr[i] == el) {
      removed = true;
    } else {
```

```

        res.push(arr[i]);
    }
}

return res;
};

class State {
    constructor(numbers, operators, edges = []) {
        this.numbers = numbers.sort((a, b) => a - b);
        this.operators = operators.sort();
        this.edges = edges;
    }

    appendEdge(state) {
        this.edges.push(state);
    }

    static toString(state) {
        return state.numbers.join(",") + "|" + state.operators.join(",");
    }

    static fromString(str) {
        str = str.split("|");

        const numbers = str[0].split(",").map(Number);
        const operators = str[1].split(",");

        return new State(numbers, operators);
    }
}

const states = new Map();

const explore = state => {
    const numbers = unique(state.numbers);
    const operators = unique(state.operators);

    for (let i = 0; i < numbers.length; i++) {
        for (let j = i; j < numbers.length; j++) {
            for (let k = 0; k < operators.length; k++) {
                const a = numbers[i];
                const b = numbers[j];
                const operator = operators[k];

                const exp = [a, operator, b].join("");
                const value = eval(exp);

                let newState = new State(
                    [...state.numbers, value],
                    removeOne(state.operators, operator)
                );
                const stateString = State.toString(newState);

```

```

        if (states.has(stateString)) {
            newState = states.get(stateString);
        } else {
            states.set(stateString, newState);
        }

        state.appendEdge(newState);

        explore(newState);
    }
}
};

const initial = new State([2, 4, 5, 7], ["+", "+", "*", "*"]);

explore(initial);

let possible = new Set();

Array.from(states.values()).forEach(state => {
    state.numbers.forEach(number => possible.add(number));
});

possible = new Set(
    Array.from(possible)
        .filter(num => num >= 200 && num <= 300)
        .sort((a, b) => a - b)
);

const impossible = Array.from(
    { length: 101 },
    (item, index) => 200 + index
).filter(item => !possible.has(item));

console.log(impossible);

```

Výsledok

```

[wiki@wiki hw5]$ node index.js
[ 213, 237, 239, 241, 281, 283 ]

```

Z intervalu <200; 300> sa pomocou daných čísel a operátorov nedajú zostaviť práve čísla:

213, 237, 239, 241, 281, 283