

Aufgabe 3: Wortsuche-Dokument

Team-ID: 00070

Team: Gaußgamz

Bearbeiter/-innen dieser Aufgabe:

Max Humbert, Nils Stäcker

1. Oktober 2021

Lösungsidee	1
Umsetzung	1
Beispiele	2
Quellcode	4

Lösungsidee

Für die Aufgabe haben wir uns drei verschiedene Schwierigkeitsstufen ausgedacht. In der einfachsten Schwierigkeitsstufe können die Wörter entweder vertikal oder horizontal im Wortgitter verteilt werden. Die Ausrichtung der Wörter ist hierbei nur von links nach rechts bzw. von oben nach unten möglich. In der zweiten Schwierigkeitsstufe werden die Wörter dann noch gespiegelt, so dass die Wörter nun auch von rechts nach links bzw. von unten nach oben vorkommen können. In unserer schwersten Schwierigkeitsstufe werden nun die Wörter auch noch diagonal ins Gitter eingefügt. Hierbei ist es möglich, dass die Wörter von links oben nach rechts unten oder von rechts oben nach links unten vorkommen. Auch können die Wörter dabei zusätzlich gespiegelt sein.

Umsetzung

Um das Wortgitter dazustellen, haben wir ein zweidimensionales Char-Array erstellt, welches wir zunächst mit Platzhaltern gefüllt hatten. Bevor wir mit dem Einfügen der Wörter begonnen haben, haben wir erstmal die Textdateien mithilfe von der Scanner und File Klasse eingelesen und in eine ArrayList gespeichert. Danach haben wir die Wörter der Länge nach sortiert, beginnend mit dem längsten Wort. Das Verteilen der Wörter, sowie das Auswählen, ob ein Wort horizontal, vertikal oder diagonal eingefügt wird, passiert völlig zufällig. Die Zufälle erzeugen wir mithilfe der Random Klasse, die uns eine Zufallszahl generiert. Im schwersten Schwierigkeitsgrad legen wir als erstes fest, ob wir das Wort spiegeln wollen (fällt bei Level 1 weg, da der Teil dafür zuständig ist, dass es auch Wörter von rechts nach links oder von unten nach oben gibt). Danach wird wieder mit einer Zufallszahl bestimmt, ob das Wort horizontal, vertikal oder diagonal eingefügt wird. Wo es eingefügt wird, hängt wiederum auch vom Zufall ab. Nachdem alle Wörter eingefügt wurden, haben wir die restlichen Platzhalter in unserem Array durch zufällige Buchstaben aus dem Alphabet ersetzt.

Beispiele

Schwierigkeitsstufe

1

0	1	2	3	4	5	6	0	1	2	3	4	5	6
1	H	X	N	E	I	N	1	P	Q	E	F	R	B
2	V	N	J	J	E	D	2	D	T	E	I	N	Q
3	I	D	U	Y	R	W	3	U	I	D	D	D	D
4	N	R	B	P	U	A	4	M	N	A	G	N	U
5	F	V	F	A	N	Z	5	J	F	C	B	E	N
6	O	H	R	F	D	A	6	T	O	X	K	R	D

Hier sind mehrere Beispiele für eine Lösung unseres Algorithmus von „worte1.txt“ mit der Schwierigkeitsstufe 1. Die Wörter sind nur von links nach rechts oder von oben nach unten verteilt.

Schwierigkeitsstufe 2

0	1	2	3	4	5	6	7	8	9	0	0	1	2	3	4	5	6	7	8	9	0
1	B	M	T	S	S	R	V	A	F	M	1	E	O	K	E	E	P	D	Y	G	N
2	I	Y	B	S	U	U	B	R	E	Z	2	B	W	R	Q	H	L	N	S	D	O
3	L	Y	X	M	L	T	P	E	S	M	3	C	O	M	P	U	T	E	R	Y	D
4	D	B	N	M	L	A	G	T	T	A	4	N	U	U	V	J	S	M	S	N	F
5	S	P	V	B	Y	T	P	U	P	U	5	L	X	D	T	N	D	M	A	U	S
6	C	E	F	G	O	S	T	P	L	S	6	B	I	L	D	S	C	H	I	R	M
7	H	N	I	I	Y	A	W	M	A	V	7	F	E	S	T	P	L	A	T	T	E
8	I	N	G	K	A	T	X	O	T	D	8	F	Q	B	S	U	W	R	I	Z	D
9	R	Q	O	W	L	J	P	C	T	D	9	R	U	T	A	T	S	A	T	M	G
0	M	B	V	A	Q	A	L	Y	E	X	0	K	M	A	J	X	V	P	W	M	L

Das sind Beispiele von „worte2.txt“ mit der Schwierigkeitsstufe 2. Hierbei sind die Wörter jetzt zusätzlich auch von rechts nach links und von unten nach oben angeordnet.

Schwierigkeitsstufe 3

0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2
1	U	Z	K	Y	N	K	L	V	W	A	L	W	M	O	N	O	G	R	A	M	M	Q
2	Q	Y	V	R	X	N	K	L	W	P	H	O	Z	X	D	L	X	Q	C	N	E	Z
3	M	J	K	D	S	O	E	L	B	B	A	K	B	E	D	P	I	I	R	X	X	Z
4	M	H	E	V	K	F	F	T	C	G	E	O	B	I	Z	P	Y	A	T	F	M	P
5	H	K	L	T	R	L	T	D	N	O	I	T	A	M	I	T	I	G	E	L	H	W
6	W	Z	B	E	J	R	R	T	R	T	T	J	L	C	H	M	P	G	X	P	J	N
7	T	V	S	Z	K	R	E	Z	O	U	D	A	J	U	C	A	P	G	J	Q	Z	M
8	J	N	S	F	J	I	X	V	P	O	T	H	Z	X	Q	T	D	G	N	F	C	G
9	C	K	O	G	J	H	M	T	O	A	E	K	D	G	V	H	O	P	V	M	V	E
0	P	S	K	I	S	C	M	Z	U	L	O	M	N	R	I	S	R	N	O	A	C	M
1	A	W	V	H	T	L	X	P	P	E	U	E	I	U	L	A	K	A	J	A	U	I
2	D	D	D	T	N	K	P	Z	N	O	E	T	N	S	J	S	A	G	H	M	G	W
3	V	H	E	D	M	H	E	D	X	H	Q	P	I	O	S	N	C	Z	W	K	D	X
4	B	K	K	R	X	S	F	F	S	B	N	K	A	O	W	I	O	V	S	P	H	C
5	E	F	O	M	E	W	Y	E	N	O	I	D	J	D	N	T	O	K	S	A	O	S
6	W	M	R	X	P	R	Q	L	I	I	P	V	L	I	L	Q	E	N	T	N	R	X
7	I	S	A	Q	K	V	I	T	F	B	U	S	E	Y	K	Q	Z	T	I	E	D	Q
8	C	X	T	U	S	O	I	U	F	I	V	I	G	E	D	Y	A	W	V	L	K	W
9	D	M	I	Z	Y	U	V	X	H	X	H	M	X	W	F	R	S	O	A	R	I	D
0	M	P	O	B	T	J	R	N	E	T	F	B	O	L	E	I	E	B	U	P	N	X
1	R	D	N	N	D	X	Z	B	A	R	Y	G	I	F	P	Z	O	E	U	Y	O	F
2	H	V	I	V	Q	A	U	P	P	V	H	N	E	G	L	U	E	S	M	S	R	Q
3	U	C	A	S	G	E	M	Y	A	D	Z	R	Q	E	F	E	U	R	M	Y	H	S
4	T	P	Q	J	V	E	L	B	Y	C	T	L	O	X	A	Y	R	F	F	F	C	V

Das letzte Beispiel ist ein Beispiel mit „worte3.txt“ mit der 3. Schwierigkeitsstufe. Hierbei werden die Wörter auch diagonal im Feld verteilt. In diesem Beispiel ist es Zufall, dass 8 von 12 Wörtern diagonal sind.

Quellcode

Der Code zum Einlesen der Datei:

```
private void einlesen(String input){
    File file = new File(input);
    lines = new ArrayList<String>();
    try{
        Scanner scanner = new Scanner(file);
        while(scanner.hasNext()){
            lines.add(scanner.nextLine());
        } catch(FileNotFoundException e){
            e.printStackTrace();
        }

        parts = lines.get(0).split(" ");
    }
}
```

Hier ein Beispiel zum horizontalen Einfügen der Wörter bei Level 2:

```
private boolean horiEinfg(String wort){
    boolean passt = true;
    char bst[] = wort.toCharArray();
    Random random = new Random();

    row = random.nextInt((zeile - bst.length + 1));
    coloum = random.nextInt(spalte);
    while(passt){
        for(int i = 0; i < bst.length; i++){
            if(Gitter[row+i][coloum] == '#' || Gitter[row+i][coloum] == bst[i]){
                passt = true;
            }else{
                passt = false;
                break;
            }
        }
        break;
    }

    if(passt == true){
        for(int i = 0; i < bst.length; i++){
            Gitter[row+i][coloum] = bst[i];
        }

        return true;
    } else{
        return false;
    }
}
```

```
}
```

Und noch der Code, der unser Level 3 ausführt:

```
public void level3(){
    reset();
    Random random = new Random();

    for(int i = 2; i < (Integer.parseInt(lines.get(1)) + 2);
i++){
        for(int j = 0; j < 50; j++){
            int zufall2 = random.nextInt(2);
            if(zufall2==0){
                int zufall = random.nextInt(4);
                if(zufall == 0){
                    if(vertEinfg(lines.get(i)) == true){
                        break;
                    }
                } else if(zufall == 1){
                    if(horiEinfg(lines.get(i)) == true){
                        break;
                    }
                }
            }else{
                if(diagonalEinfg(lines.get(i)) == true){
                    break;
                }
            }
        }
        }else{
            drehen(i);
            int zufall = random.nextInt(4);
            if(zufall == 0){
                if(vertEinfg(lines.get(i)) == true){
                    break;
                }
            } else if(zufall == 1){
                if(horiEinfg(lines.get(i)) == true){
                    break;
                }
            }
        }else{
            if(diagonalEinfg(lines.get(i)) == true){
                break;
            }
        }
    }
}

}
```

```
        randomAuffüllen();  
        zeigen();  
    }
```