



Αναπαράσταση προβλήματος

Για την αναπαράσταση του προβλήματος χρησιμοποιούμε ένα string για κάθε μπουκάλι. Κάθε string αποτελείται από 0 έως 4 χαρακτήρες, όπου κάθε χαρακτήρας αντιστοιχεί σε ένα χρώμα. Για την αναπαράσταση όλων των μπουκαλιών χρησιμοποιούμε έναν δυναμικό πίνακα (`vector<string> bottles`) με μέγεθος ίσο με το πλήθος των μπουκαλιών. Ο πίνακας `bottles` είναι private μέλος της κλάσης `Bottles`, κάθε αντικείμενο της οποίας αποτελεί ένα στιγμιότυπο του προβλήματος.

Για παράδειγμα μερικά στιγμιότυπα ενός τυχαίου προβλήματος με 6 μπουκάλια είναι τα εξής:

Αρχική κατάσταση: {"cdac", "dadc", "bacb", "dabb", "", ""}
Ενδιάμεση κατάσταση: {"c", "dadc", "bacc", "dd", "bbb", "aa"}
Τελική κατάσταση (Λύση): {"cccc", "", "", "dddd", "bbbb", "aaaa"}

	Char	Color
1	A	Aqua
2	B	Brown
3	C	Coral
4	D	Dandelion
5	E	Ecru
6	F	Fuchsia
7	G	Green
8	H	Honey
9	I	Indigo
10	J	Jade
11	K	Khaki
12	L	Lavender
13	M	Maroon
14	N	Navy blue
15	O	Orange
16	P	Pink
17	Q	Quartz grey
18	R	Red
19	S	Scarlet
20	T	Turquoise

Συγκεκριμένα, η κλάση **Bottles** έχει ως private μέλη τα εξής:

vector<string> bottles, ένα vector με μέγεθος ίσο με τον αριθμό των μπουκαλιών του προβλήματος, όπου κάθε κελί περιέχει ένα string με 0 έως 4 χαρακτήρες και κάθε χαρακτήρας αναπαριστά ένα χρώμα.

string actionName, ένα string με το όνομα της κίνησης που οδήγησε στο τρέχον στιγμιότυπο του προβλήματος.

Bottles *prev, ένας δείκτης που δείχνει στο προηγούμενο στιγμιότυπο του προβλήματος.

Υλοποίηση βασικών μεθόδων

Οι βασικές μέθοδοι της κλάσης Bottles είναι οι εξής:

bool isSolved()

η οποία ελέγχει αν το **vector<string> bottles** αντιστοιχεί σε λύση του προβλήματος, δηλαδή ελέγχει αν όλα τα μπουκάλια, εκτός από δύο, είναι γεμάτα με το ίδιο χρώμα το καθένα.

bool isValidMove(string sourceBottle, string destinationBottle)

η οποία επιστρέφει πόσα ml από το μπουκάλι **sourceBottle** μπορούμε να ρίξουμε στο μπουκάλι **destinationBottle**, με βάση τους κανόνες του προβλήματος. Επιστρέφει 0, αν δεν επιτρέπεται κίνηση από το ένα μπουκάλι στο άλλο.

bool makeMove(Bottles &n, string sourceBottle, string destinationBottle, int i, int j)

η οποία κάνει την κίνηση από το μπουκάλι **sourceBottle** στο μπουκάλι **destinationBottle**, αν αυτό επιτρέπεται, αποθηκεύοντας το αποτέλεσμα στο αντικείμενο **n**, το οποίο παριστάνει μια πιθανή επόμενη κατάσταση που μπορεί να προκύψει από την τρέχουσα κατάσταση του προβλήματος. Αν δεν επιτρέπεται κίνηση από το ένα μπουκάλι σε άλλο, η συνάρτηση **makeMove** επιστρέφει false.

vector <Bottles *> expand()

η οποία επιστρέφει ένα **vector<Bottles *> children** με όλες τις πιθανές καταστάσεις που μπορεί να προκύψουν από την τρέχουσα κατάσταση (σύμφωνα με τους τελεστές μετάβασης του προβλήματος) καλώντας την συνάρτηση **makeMove**.

string getKey()

η οποία επιστρέφει ένα μοναδικό string-κλειδί για το στιγμιότυπο του προβλήματος. Για να εξασφαλίσουμε ότι θα παραχθεί το ίδιο κλειδί για ισοδύναμα στιγμιότυπα του προβλήματος, δηλαδή στιγμιότυπα όπου τα μπουκάλια είναι ίδια αλλά σε άλλες θέσεις, το κλειδί αποτελεί ένα string, το οποίο παράγεται ταξινομώντας τα μπουκάλια αλφαβητικά και συγχωνεύοντάς τα σε ένα ενιαίο string χωρισμένα με «;». Για παράδειγμα, το κλειδί για το στιγμιότυπο {"cba", "dec", "ace", "cba", "eaeb", "bddd"} είναι το "ace;b;cba;cba;ddd;dec;eaeb;").

Στη main έχουμε τις συναρτήσεις:

Bottles *BFS(Bottles *initial, long long &examined, long long &mem)

η οποία υλοποιεί τον αλγόριθμο BFS και επιστρέφει ένα δείκτη **Bottles *s** με τη λύση του προβλήματος ή **nullptr** αν δεν βρεθεί λύση.

string random_string(int k)

η οποία επιστρέφει ένα string, που αποτελείται από k διαφορετικούς χαρακτήρες (k πρώτα γράμματα της αλφαβήτας) και ο κάθε χαρακτήρας εμφανίζεται τυχαία από 4 φορές μέσα στο string

string *random_bottles(int numberOfBottles)

η οποία επιστρέφει έναν πίνακα με numberOfBottles κελιά, όπου τα numberOfBottles μείον δύο πρώτα κελιά έχουν 4 τυχαίους χαρακτήρες το καθένα με βάση το string που θα επιστρέψει η **random_string**.

Για παράδειγμα, αν η main καλέσει την **random_bottles** με numberOfBottles=7, τότε η **random_string** θα δημιουργήσει τυχαία το string "cbabdecdbacedcbadeaeb", με χαρακτήρες από το "a" έως το "e" (5 χαρακτήρες) και η **random_bottles** θα δημιουργήσει τον πίνακα:

cbab	decdb	aced	cbad	eaeb		
------	-------	------	------	------	--	--

Παραδοχές

Το πρόγραμμα δημιουργεί τυχαία μια αρχική κατάσταση για το πρόβλημα. Για να το πετύχει αυτό, επιλέγει τυχαία τον αριθμό των μπουκαλιών με εύρος από 4 (για να υπάρχουν τουλάχιστον 2 χρώματα) έως 20 μπουκάλια. Με βάση των αριθμό των μπουκαλιών, δημιουργεί τυχαία την αρχική κατάσταση καλώντας την συνάρτηση **random_bottles**, εξασφαλίζοντας πως θα υπάρχουν αρχικά δύο άδεια μπουκάλια και 4ml από κάθε χρώμα διασκορπισμένα στα υπόλοιπα μπουκάλια.

Αποτελέσματα προγράμματος για 3 τυχαίες εκτελέσεις

Παρακάτω, φαίνεται η εκτέλεση του προγράμματος για 5 μπουκάλια, δηλαδή 3 διαφορετικά χρώματα.

```
Number of bottles: 5

BFS: depth = 9, Mem: 319, Examined: 216

Starting from:          acac;cbba;bacb;____;____;
Pour 1 ml from 1 to 4;  aca_;cbba;bacb;c____;____;
Pour 1 ml from 2 to 1;  acaa;cbb_;bacb;c____;____;
Pour 1 ml from 3 to 2;  acaa;cbbb;bac_;c____;____;
Pour 1 ml from 3 to 4;  acaa;cbbb;ba__;cc__;____;
Pour 2 ml from 1 to 3;  ac__;cbbb;baaa;cc__;____;
Pour 1 ml from 1 to 4;  a___;cbbb;baaa;ccc_;____;
Pour 3 ml from 3 to 1;  aaaa;cbbb;b___;ccc_;____;
Pour 3 ml from 2 to 3;  aaaa;c___;bbbb;ccc_;____;
Pour 1 ml from 2 to 4;  aaaa;____;bbbb;cccc;____;
Time elapsed: 0.019 seconds
```

Παρακάτω φαίνεται η εκτέλεση του προγράμματος για 12 μπουκάλια, δηλαδή 10 διαφορετικά χρώματα.

```
Number of bottles: 12

BFS: depth = 29, Mem: 21352, Examined: 20566

Starting from:      jjdf;iggb;cebh;egai;ecib;fddc;ghaa;aifh;hfjc;ebdj;____;____;
Pour 1 ml from 2 to 11; jjdf;igg_;cebh;egai;ecib;fddc;ghaa;aifh;hfjc;ebdj;b____;____;
Pour 1 ml from 5 to 11; jjdf;igg_;cebh;egai;eci_;fddc;ghaa;aifh;hfjc;ebdj;bb____;____;
.
.
Pour 1 ml from 10 to 8;  jjjj;iii;cccc;____;eee_;ffff;gggg;dddd;hhhh;eb____;bbb_;aaaa;
Pour 1 ml from 10 to 11; jjjj;iii;cccc;____;eee_;ffff;gggg;dddd;hhhh;e____;bbb;aaaa;
Pour 3 ml from 5 to 10;  jjjj;iii;cccc;____;____;ffff;gggg;dddd;hhhh;eeee;bbb;aaaa;
Time elapsed: 4.454 seconds
```

Παρακάτω φαίνεται η εκτέλεση του προγράμματος για 19 μπουκάλια, δηλαδή 17 διαφορετικά χρώματα.

```
Number of bottles: 19

BFS: depth = 56, Mem: 66302, Examined: 66051

Starting from:
jada;lkjj;eppq;qolf;gbfd;aodp;ebgq;ephi;fcch;enoq;knmj;fnlb;gdib;konl;chih;gikc;mamm;____;____;
Pour 1 ml from 3 to 18;
jada;lkjj;epp_;qolf;gbfd;aodp;ebgq;ephi;fcch;enoq;knmj;fnlb;gdib;konl;chih;gikc;mamm;q____;____;
Pour 1 ml from 6 to 3;
jada;lkjj;eppp;qolf;gbfd;aod_;ebgq;ephi;fcch;enoq;knmj;fnlb;gdib;konl;chih;gikc;mamm;q____;____;
.
.
Pour 1 ml from 12 to 2;
jjjj;llll;bbbb;nnn_;pppp;aaaa;oooo;____;fff_;eeee;iii;fn____;hhhh;kkkk;cccc;gggg;dddd;qqqq;mmmm;
Pour 1 ml from 12 to 4;
jjjj;llll;bbbb;nnnn;pppp;aaaa;oooo;____;fff_;eeee;iii;f____;hhhh;kkkk;cccc;gggg;dddd;qqqq;mmmm;
Pour 3 ml from 9 to 12;
jjjj;llll;bbbb;nnnn;pppp;aaaa;oooo;____;____;eeee;iii;ffff;hhhh;kkkk;cccc;gggg;dddd;qqqq;mmmm;
Time elapsed: 40.37 seconds
```

Συνοπτικά:

Number of bottles	Depth	Memory	Examined	Time
5	9	319	216	0.019 seconds
12	29	21352	20566	4.454 seconds
19	56	66302	66051	40.37 seconds