



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
"МИРЭА - Российский технологический университет"

РТУ МИРЭА

Институт информационных технологий
Кафедра Корпоративных информационных систем

**ОТЧЕТ
ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 2
по дисциплине
«Конфигурационное управление»**

Выполнил студент группы ИКБО-12-22

Карабанов С.Е.

Принял
преподаватель

Емельянов А. М.

Практическая работа выполнена « 25 » 09 2023 г.

(подпись студента)

«Зачтено» «__»__ 2023 г.

(подпись руководителя)

Москва 2023

СОДЕРЖАНИЕ

ЗАДАЧА 1	3
ЗАДАЧА 2	4
ЗАДАЧА 3	6
ЗАДАЧА 4	10
ЗАДАЧА 5	11
ЗАДАЧА 6	13
ЗАДАЧА 7	14
СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ.....	15

ЗАДАЧА 1

Постановка задачи:

Вывести служебную информацию о пакете matplotlib (Python). Разобрать основные элементы содержимого файла со служебной информацией из пакета. Как получить пакет без менеджера пакетов, прямо из репозитория?

Решение задачи:

Из этого вывода можно узнать название, версию, описание, сайт, авторов и их почты, лицензию, расположение на диске, зависимости пакета, какие пакеты зависят от данного.

Получить пакет matplotlib прямо из репозитория можно по ссылке:

<https://pypi.org/project/matplotlib/#files>

```
Name: matplotlib
Version: 3.8.0
Summary: Python plotting package
Home-page: https://matplotlib.org
Author: John D. Hunter, Michael Droettboom
Author-email: matplotlib-users@python.org
License: PSF
Location:      /home/Semyon/.local/lib/python3.10/python3.10/site-
packages
Requires:  contourpy,  cyclor,  fonttools,  kiwisolver,  numpy,
packaging, pillow, pyparsing, python-dateutil
Required-by:
```

ЗАДАЧА 2

Постановка задачи:

Вывести служебную информацию о пакете express (JavaScript). Разобрать основные элементы содержимого файла со служебной информацией из пакета. Как получить пакет без менеджера пакетов, прямо из репозитория?

Решение:

```
express@4.18.2 | MIT | deps: 31 | versions: 270
Fast, unopinionated, minimalist web framework
http://expressjs.com/
keywords: express, framework, sinatra, web, http, rest, restful,
router, app, api
dist
.tarball: https://registry.npmjs.org/express/-/express-4.18.2.tgz
.shasum: 3fabe08296e930c796c19e3c516979386ba9fd59
.integrity: sha512-
5/PsL6iGPdfQ/lKM1UuielYgv3BUoJfz1aUwU9vHZ+J7gyvwdQXFEBIEIaxeGf0GI
creATNyBExtalisDbuMqQ==
.unpackedSize: 213.9 kB
dependencies:
accepts: ~1.3.8          depd: 2.0.0          methods:
~1.1.2
array-flatten: 1.1.1      encodeurl: ~1.0.2    on-
finished: 2.4.1
body-parser: 1.20.1       escape-html: ~1.0.3  parseurl:
~1.3.3
content-disposition: 0.5.4 etag: ~1.8.1         path-to-
regexp: 0.1.7
content-type: ~1.0.4      finalhandler: 1.2.0  proxy-addr:
~2.0.7
cookie-signature: 1.0.6   fresh: 0.5.2         qs: 6.11.0
```

```
cookie: 0.5.0          http-errors: 2.0.0          range-  
parser: ~1.2.1  
  
debug: 2.6.9           merge-descriptors: 1.0.1    safe-  
buffer: 5.2.1  
  
(...and 7 more.)  
  
maintainers:  
- mikeal <mikeal.rogers@gmail.com>  
- dougwilson <doug@somethingdoug.com>  
  
dist-tags:  
latest: 4.18.2         next: 5.0.0-beta.1  
  
published 11 months ago by dougwilson <doug@somethingdoug.com>
```

Из этого вывода можно узнать название, версию, лицензию, количество зависимостей, количество версий, описание, сайт, ключевые слова, ссылку на пакет, хеш-суммы, размер в распакованном виде, зависимости, сопроводителей и их почты, теги дистрибутива: последняя версия и следующая версия, комментарий автора.

Получить пакет express прямо из репозитория можно по ссылке:
<https://registry.npmjs.org/express/-/express-4.17.1.tgz>

ЗАДАЧА 3

Постановка задачи:

Сформировать graphviz-код и получить изображения зависимостей matplotlib и express.

Решение:

Graphviz-код для matplotlib

```
digraph matplotlib {  
matplotlib -> contourpy;  
matplotlib -> cycler;  
matplotlib -> dateutil;  
matplotlib -> fontTools;  
matplotlib -> kiwisolver;  
matplotlib -> NumPy;  
matplotlib -> packaging;  
matplotlib -> pyparsing;  
matplotlib -> Pillow;  
matplotlib -> "importlib-resources";  
}
```

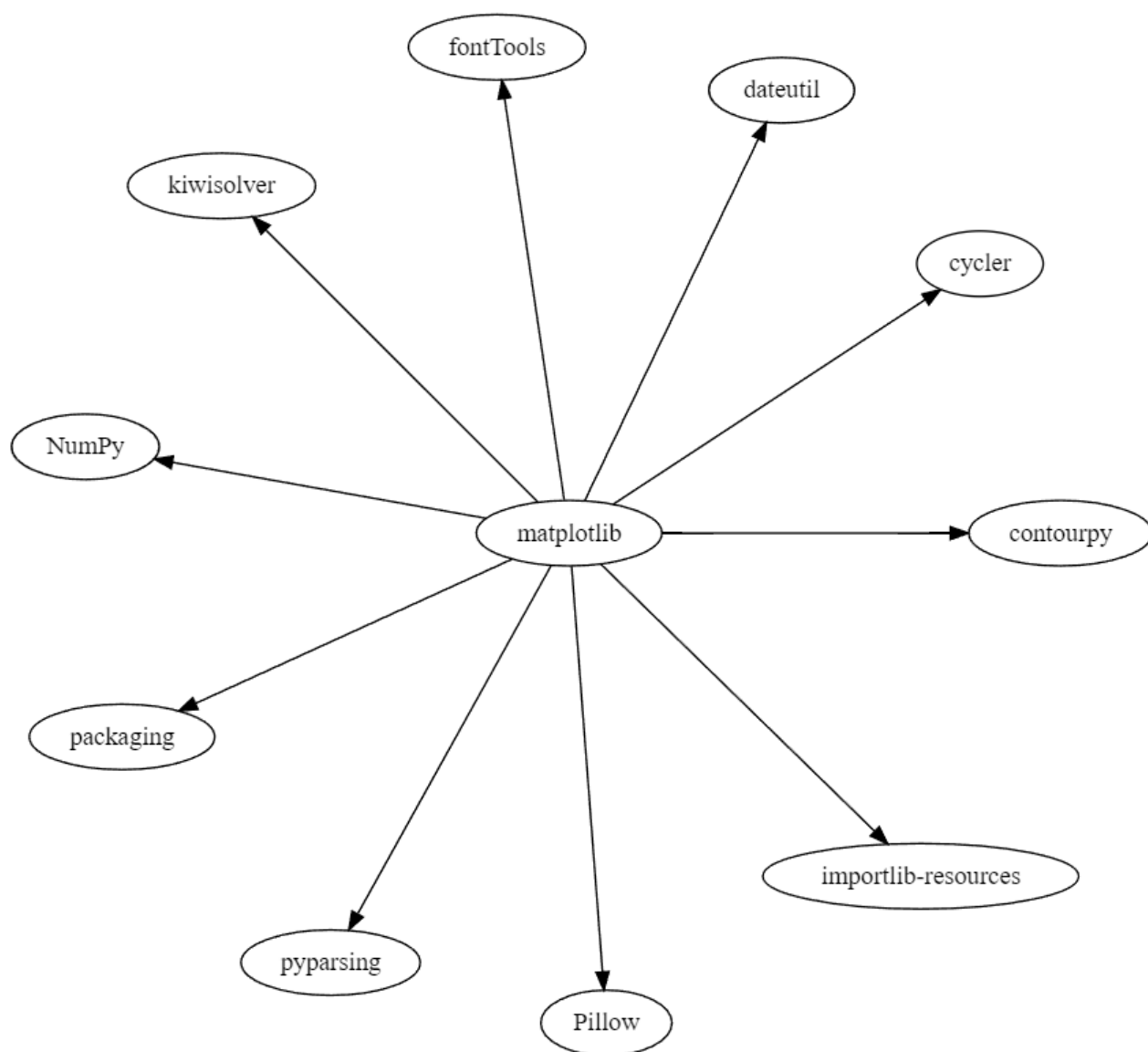


Рисунок 1 – Визуализация зависимостей пакета matplotlib

Graphviz-код для express

```
digraph express {  
  express -> accepts;  
  express -> array_flatten;  
  express -> "body-parser";  
  express -> "content-disposition";  
  express -> "content-type";  
  express -> "cookie-signature";  
  express -> cookie;  
  express -> debug;  
  express -> depd;  
  express -> encodeurl;  
  express -> "escape-html";  
  express -> etag;  
  express -> finalhandler;  
  express -> fresh;  
  express -> "merge-descriptors";  
  express -> methods;  
  express -> "on-finished";  
  express -> parseurl;  
  express -> "path-to-regexp";  
  express -> "proxy-addr";  
  express -> qs;  
  express -> "range-parser";  
  express -> "safe-buffer";  
  express -> send;  
  express -> "serve-static";  
  express -> setprototypeof;  
  express -> statuses;  
  express -> "type-is";  
  express -> "utils-merge";  
  express -> vary;  
}
```




Рисунок 2 – Визуализация зависимостей пакета express

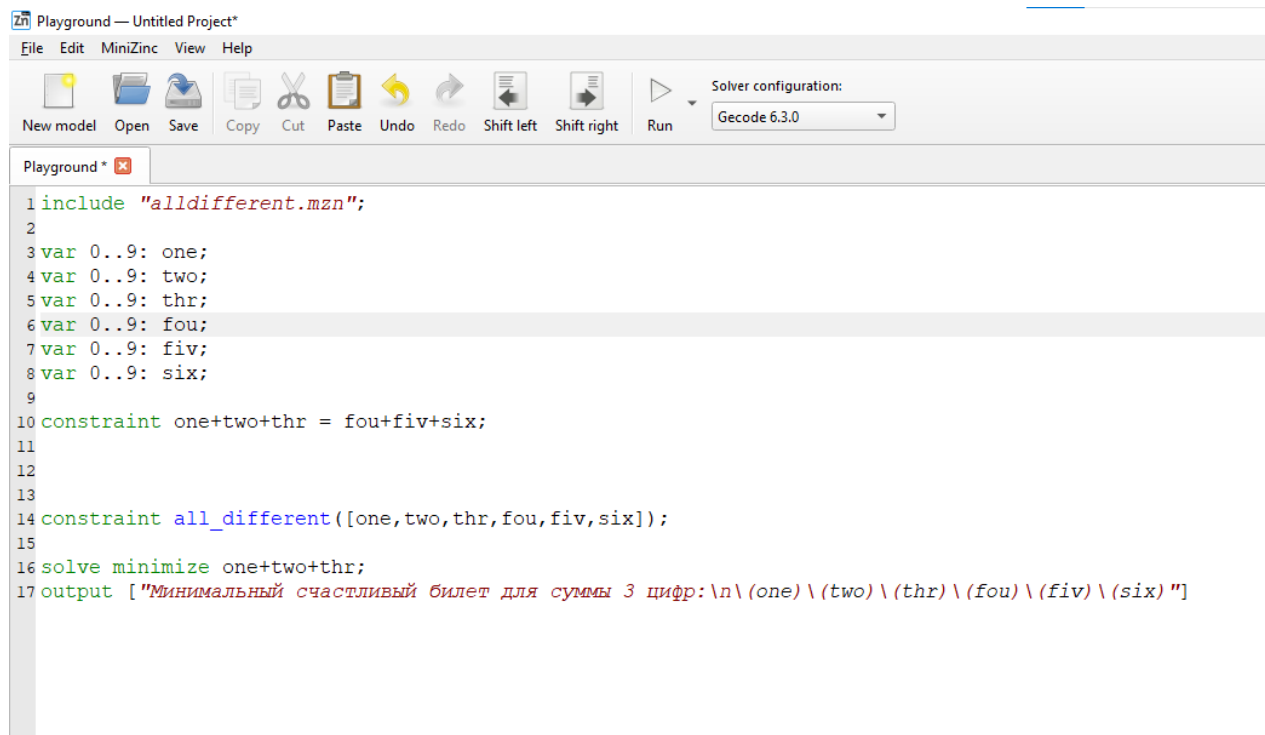
ЗАДАЧА 4

Постановка задачи:

Изучить основы программирования в ограничениях. Установить MiniZinc, разобраться с основами его синтаксиса и работы в IDE.

Решить на MiniZinc задачу о счастливых билетах. Добавить ограничение на то, что все цифры билета должны быть различными (подсказка: используйте `all_different`). Найти минимальное решение для суммы 3 цифр.

Решение:



```
1 include "alldifferent.mzn";
2
3 var 0..9: one;
4 var 0..9: two;
5 var 0..9: thr;
6 var 0..9: fou;
7 var 0..9: fiv;
8 var 0..9: six;
9
10 constraint one+two+thr = fou+fiv+six;
11
12
13
14 constraint all_different([one,two,thr,fou,fiv,six]);
15
16 solve minimize one+two+thr;
17 output [ "Минимальный счастливый билет для суммы 3 цифр:\n(one)\(two)\(thr)\(fou)\(fiv)\(six) " ]
```

Рисунок 3 – код для решения задачи о счастливых билетах в MiniZinc

```
Минимальный счастливый билет для суммы 3 цифр:
810432
-----
Минимальный счастливый билет для суммы 3 цифр:
620431
-----
=====
```

Рисунок 4 – результат выполнения

ЗАДАЧА 5

Постановка задачи:

Решить на MiniZinc задачу о зависимостях пакетов для рисунка, приведенного ниже.

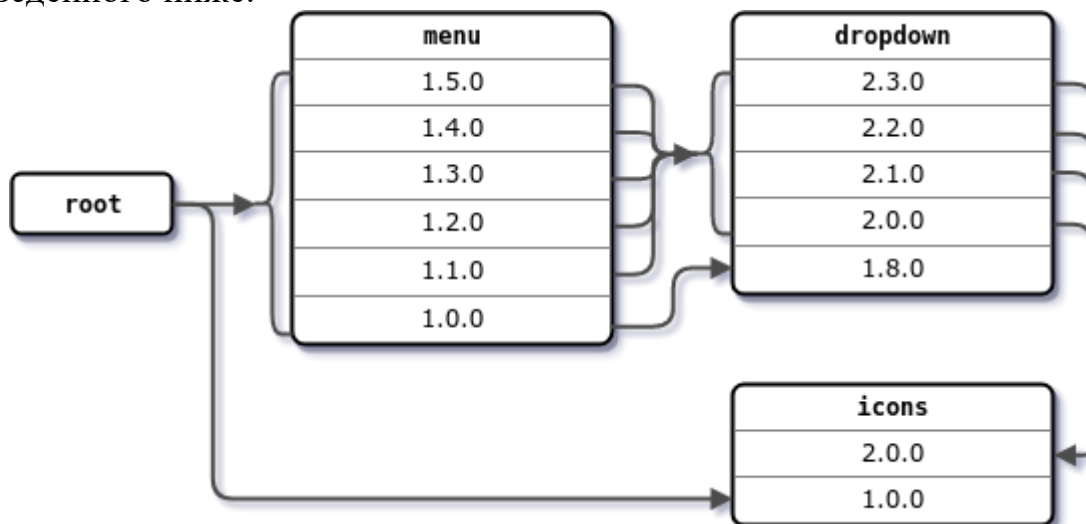


Рисунок 5 – условие задачи 5

Решение:

```
int: rootMajor = 1;
int: patch = 0;
var 0..1: MenuMajor;
var 0..5: MenuMinor;
var 0..2: DropdownMajor;
var 0..8: DropdownMinor;
var 0..2: IconsMajor;
constraint if rootMajor = 1 then MenuMajor > 0 /\ MenuMinor >= 0
/\ IconsMajor = 1 endif;
constraint if MenuMinor > 0 then DropdownMajor = 2 /\ DropdownMinor
<= 3 else DropdownMajor = 1 /\ DropdownMinor = 8 endif;
constraint if DropdownMajor = 2 then IconsMajor = 2 endif;

solve satisfy;
output["Для установки root пакета необходимы:\nmenu
\(MenuMajor).\ (MenuMinor).\ (patch)\ndropdown
\(DropdownMajor).\ (DropdownMinor).\ (patch)\nicons
\(IconsMajor).\ (patch).\ (patch)"]
```

Рисунок 6 – код для решения задачи 5

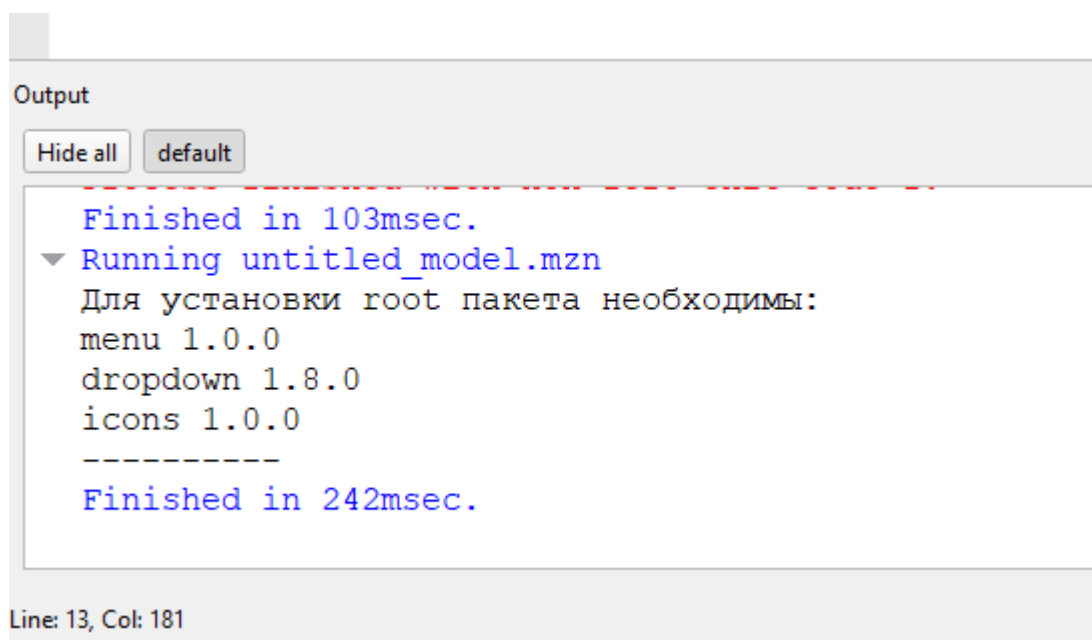


Рисунок 7 – результат выполнения кода для задачи 5

ЗАДАЧА 6

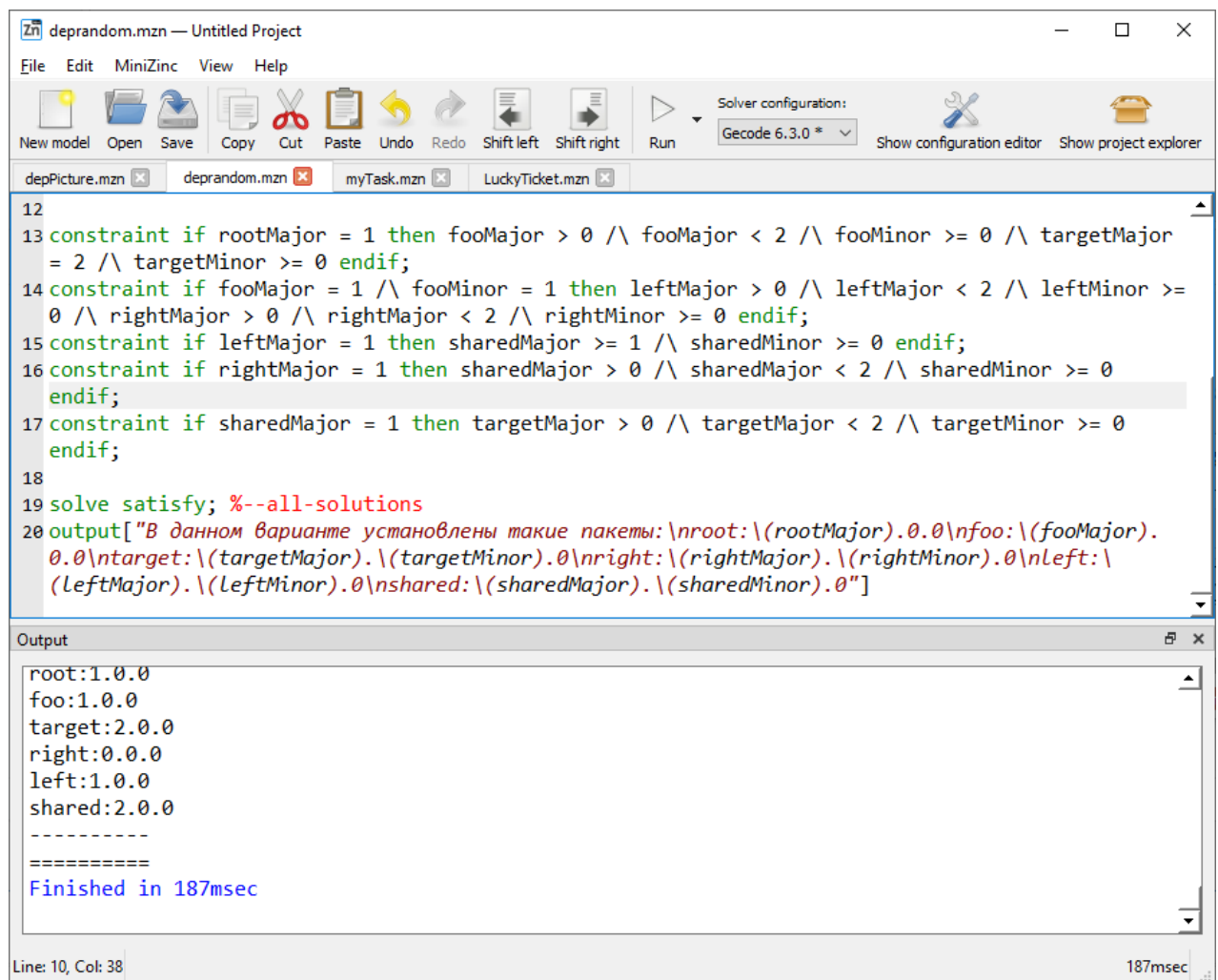
Постановка задачи:

Решить на MiniZinc задачу о зависимостях пакетов для следующих данных:

```
root 1.0.0 зависит от foo ^1.0.0 и target ^2.0.0.
foo 1.1.0 зависит от left ^1.0.0 и right ^1.0.0.
foo 1.0.0 не имеет зависимостей.
left 1.0.0 зависит от shared >=1.0.0.
right 1.0.0 зависит от shared <2.0.0.
shared 2.0.0 не имеет зависимостей.
shared 1.0.0 зависит от target ^1.0.0.
target 2.0.0 и 1.0.0 не имеют зависимостей.
```

Рисунок 8 – условия задачи 6

Решение:



The screenshot shows the MiniZinc IDE interface. The main editor displays a MiniZinc model file named `deprandom.mzn`. The code defines constraints for package dependencies and uses the `solve satisfy` command to find a solution. The output window shows the resulting package versions: `root:1.0.0`, `foo:1.0.0`, `target:2.0.0`, `right:0.0.0`, `left:1.0.0`, and `shared:2.0.0`. The solver finished in 187ms.

```
12
13 constraint if rootMajor = 1 then fooMajor > 0 /\ fooMajor < 2 /\ fooMinor >= 0 /\ targetMajor
   = 2 /\ targetMinor >= 0 endif;
14 constraint if fooMajor = 1 /\ fooMinor = 1 then leftMajor > 0 /\ leftMajor < 2 /\ leftMinor >=
   0 /\ rightMajor > 0 /\ rightMajor < 2 /\ rightMinor >= 0 endif;
15 constraint if leftMajor = 1 then sharedMajor >= 1 /\ sharedMinor >= 0 endif;
16 constraint if rightMajor = 1 then sharedMajor > 0 /\ sharedMajor < 2 /\ sharedMinor >= 0
   endif;
17 constraint if sharedMajor = 1 then targetMajor > 0 /\ targetMajor < 2 /\ targetMinor >= 0
   endif;
18
19 solve satisfy; %--all-solutions
20 output["В данном варианте установлены такие пакеты:\nroot:\(rootMajor).0.0\nfoo:\(fooMajor).
   0.0\ntarget:\(targetMajor).\ \(targetMinor).0\nright:\(rightMajor).\ \(rightMinor).0\nleft:\
   \(leftMajor).\ \(leftMinor).0\nshared:\(sharedMajor).\ \(sharedMinor).0"]
```

Output

```
root:1.0.0
foo:1.0.0
target:2.0.0
right:0.0.0
left:1.0.0
shared:2.0.0
-----
=====
Finished in 187msec
```

Line: 10, Col: 38 187msec

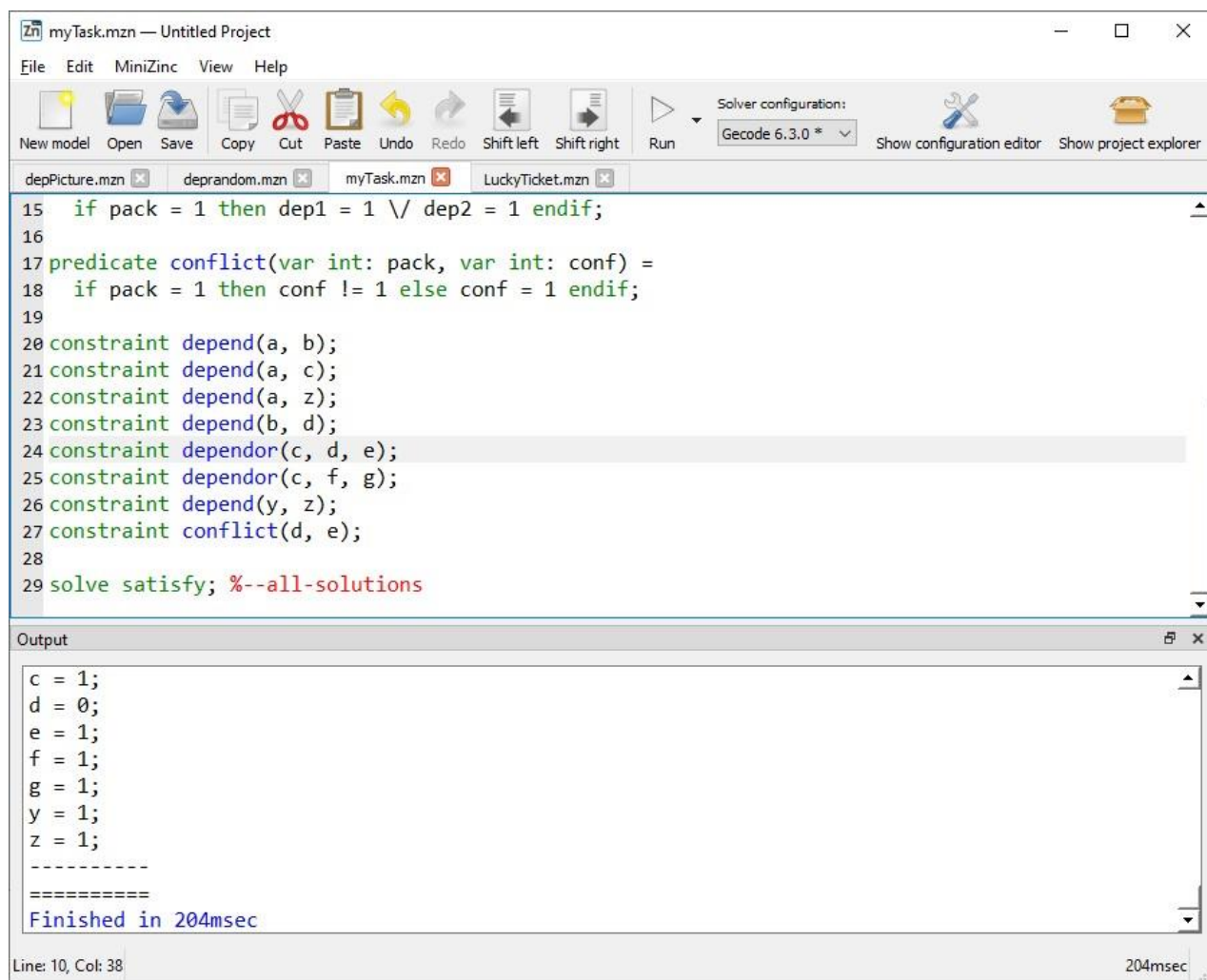
Рисунок 9 – решение задачи 6

ЗАДАЧА 7

Постановка задачи:

Представить на MiniZinc задачу о зависимостях пакетов в общей форме, чтобы конкретный экземпляр задачи описывался только своим набором данных.

Решение:



The screenshot shows the MiniZinc IDE interface. The top menu bar includes File, Edit, MiniZinc, View, and Help. The toolbar contains icons for New model, Open, Save, Copy, Cut, Paste, Undo, Redo, Shift left, Shift right, Run, and Solver configuration (set to Gecode 6.3.0 *). The main editor window displays the file myTask.mzn with the following code:

```
15 if pack = 1 then dep1 = 1 \ / dep2 = 1 endif;
16
17 predicate conflict(var int: pack, var int: conf) =
18   if pack = 1 then conf != 1 else conf = 1 endif;
19
20 constraint depend(a, b);
21 constraint depend(a, c);
22 constraint depend(a, z);
23 constraint depend(b, d);
24 constraint dependor(c, d, e);
25 constraint dependor(c, f, g);
26 constraint depend(y, z);
27 constraint conflict(d, e);
28
29 solve satisfy; %--all-solutions
```

The Output window at the bottom shows the solution:

```
c = 1;
d = 0;
e = 1;
f = 1;
g = 1;
y = 1;
z = 1;
-----
=====
Finished in 204msec
```

The status bar at the bottom indicates "Line: 10, Col: 38" and "204msec".

Рисунок 10 – решение задачи 7

СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ

1. Конфигурационное управление. Учебное пособие. Курс в РТУ МИРЭА // П.Н. Советов: [сайт] – 2021. – URL: <https://github.com/true-grue/kisscm> (дата обращения: 25.09.2023).