

Reyhane Shahrokhian 99521361

HomeWork5 of Compiler Course

Dr. Parsa

I changed 3 files: grammar, code generator, and listener.

- Grammar

I added the hint option in the wanted format to the grammar file, which can be either True or False.

```
program: output? hints? initiate_game bomb_placements;
```

```
hints: 'hint:' hint_option;  
hint_option: 'True' | 'False';
```

- Code generator

First, in the initialization part I added hints to the non_operands list and initialized a hints attribute to False.

```
def __init__(self):  
    self.non_operands = ['program', 'initiate_game',  
                        'bomb_location', 'output',  
                        'bomb_placements', 'begin_scope_operator',  
                        'end_scope_operator', 'hints']  
    self.operand_stack = []  
    self.code_stack = []  
    self.hints = False
```

Then, I write a set_hints() function to set the hints attribute based on the parsed value and a generate_hint_code() function to calculate the number of bombs adjacent to each cell.

```
def set_hints(self):
    hint_option = self.operand_stack.pop()
    self.hints = (hint_option == 'True')
```

Then, I modified the `generate_program()` function to output the bomb board with or without hints based on the `hints` attribute.

The hints_code is the output of the generate_hint_code() function.

```
if self.hints:
    hints_code = self.generate_hint_code()
else:
    hints_code = ''
```

- Listener

First, in the initialization part I added hints to the overridden_rules list to handle this rule explicitly.

```
self.overridden_rules = ['program', 'initiate_game', 'output', 'hints']
```

Then, I added an exitHints() method to properly handle the hints rule and create the corresponding AST node.

```
def exitHints(self, ctx):
    make_ast_subtree(self.ast, ctx, "hints", keep_node=True)
```

Input:

```
output: console
game: 5 X 8
bomb: 1 , 2
bomb: 4 , 6
bomb: 5 , 8
```

Output:

```
*****
*****
*****
*****
*****
*****
```

AST:



```

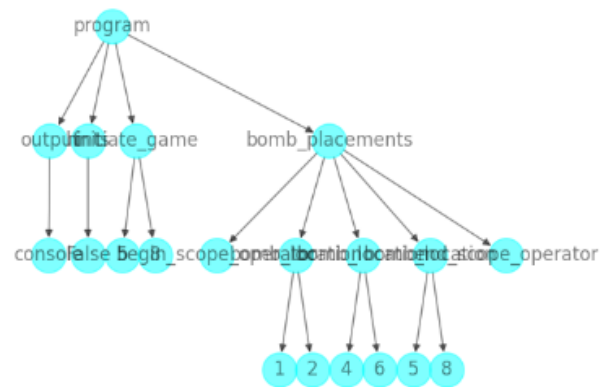
output: console
hint: True
game: 5 X 8
bomb: 1 , 2
bomb: 4 , 6
bomb: 5 , 8

```

```

#*#####
#####
#####
#####*##
#####*

```



```

output: console
hint: False
game: 5 X 8
bomb: 1 , 2
bomb: 4 , 6
bomb: 5 , 8

```

```

1*1#####
111#####
####111#
####1*21
####112*

```

