Question 1

Transitive verbs are action verbs with a direct object, whereas those without a direct object are considered to be intransitive. In the grammar in *g1* in *gay.py*, I replaced all the 'V' corresponding to verbs with 'IV' and 'TV', wherein, the former refers to intransitive verbs and the latter to transitive verbs. Given that intransitive verbs do not take DP-complements and CP complements, for the categorical rules, I discarded rules such as:

- *VP -> V + CP*

- *VP -> V + DP*

- *VP -> V + DP + PP*

- *VP -> V + DP + CP*

- *VP -> V + DP + VP*

in the case where, 'V' is replaced by 'IV'. Furthermore, in the lexical rules, I classified the given six verbs into 'IV' and 'TV'. This ensures that sentences like * *Sue laughs the student* are not accepted. Considering an object should follow a transitive verb, I replaced the rule *VP -> V* with *VP -> IV*, as transitive verbs can not be at the end of a sentence.

```
>>> from gay import *
>>> from td import *
>>> recognize(g1, "Sue laughs the student")
False
```

The above code snippet displays the output as obtained for the given sentence.

Question 2

In order to build a minimalistic grammar to ensure the given rules on the left are accepted by the parser, and the ones on the right are not accepted, the following changes to *g1* were adopted:

- the verb 'V' gives verb in base form (VB), verb in past tense (VBD), verb in present participle (VBG), verbs in third person singular present (VBZ), verbs in past participle (VBN), verb 'have' in third person singular present (VBN) and modal auxillary verbs (VM).

- the given action verbs in original g1 are modified in the lexical rules so as to fit in the above-defined categories

- 'be' is introduced as 'VB', i.e, base form of the word 'be'; 'is' as the third person singular present of the verb 'be'; 'been' as the past participle of the verb 'be'; 'has' as the third person singular present of the verb 'have'; and 'has' as the modal auxillary verbs.

- 'laughed' and 'laughing' are classified as 'VBD' and 'VBG' respectively.

- in the categorical rules, a new tag 'X1' is introduced, which maps to 'DP', 'PP', 'CP', 'DP + PP', 'DP + CP' and 'DP + VP'. This is done in order to avoid similar rule breakdowns due to the expansion over verbs.

- in the lexical rules, verb forms such as 'VBD', 'VBZ', 'VBG', 'VBN', 'VM', VHZ' are given by 'V'

In the lexical rules, rules are introduced to ensure the preservation of properties such that the auxillary verbs are only followed by 'VB', 'VBD' and 'VBG'. The rule $V \rightarrow V + VP$ is modified in order to feed in the dependency of the auxillary verbs on action verbs and rest of the verb phrases. The verbs 'have' and 'be' take participial VP complements, which is further replaced by combinations of compatible verb-forms and 'X1', the combination of which represents 'VP'. By running the code for the given sentences, the following output is obtained:

```
$ python td.py
['Sue', 'will', 'laugh'] True
['Sue', 'has', 'laughed'] True
['Sue', 'is', 'laughing'] True
['Sue', 'has', 'been', 'laughing'] True
['Sue', 'will', 'be', 'laughing'] True
['Sue', 'will', 'laughs'] False
['Sue', 'has', 'laughs'] False
['Sue', 'is', 'laughed'] False
['Sue', 'will', 'been', 'laughing'] False
['Sue', 'has', 'be', 'laugh'] False
```

It can be observed that the outputs for each sentence are as expected.