Question 1:

```
# classes we are considering for word generators:
# print("classes we are considering for word generators:")
# print("Class 1 = play, walk, talk")
# print("Class 2 = dance, mince")
# print("Class 3 = marry, carry")



                                                                    #
verb analysis
inp = input("what are u entering? verb or noun: ")
print("")
if (inp == "verb" or inp == "Verb"):
    wrd = input("Enter the word: ")
    print("")
    print("Word forms:")

    if (wrd[-1] == "e" and wrd[-2] != "i" and wrd != "leave"):      #
dance, mince, bake
        print(wrd)
        print(wrd+"s")
        print(wrd[:-1]+"ed")
        print(wrd[:-1]+"ed")
        print(wrd[:-1]+"ing")
    elif (wrd[-1] == "y" and wrd != "play" and wrd != "slay"):      #
marry, carry, cry
        print(wrd)
        print(wrd[:-1]+"ies")
        print(wrd[:-1]+"ied")
        print(wrd[:-1]+"ied")
        print(wrd+"ing")
    elif (wrd[-1] == "c"):                                          #
panic, mimic, frolic
        print(wrd)
        print(wrd+"s")
        print(wrd+"ked")
        print(wrd+"ked")
        print(wrd+"king")
    elif (wrd[-1] == "h"):                                          #
catch, watch, crouch
```

```
        print(wrd)
        print(wrd+"es")
        print(wrd+"ed")
        print(wrd+"ed")
        print(wrd+"ing")
    elif (wrd[-2] == "i" and wrd[-1] == "e" ):                      #
lie, die, tie
        print(wrd)
        print(wrd+"s")
        print(wrd+"d")
        print(wrd+"d")
        print(wrd[:-2]+"ying")

    else:                                                           #
play, walk, talk, slay
        print(wrd)
        print(wrd+"s")
        print(wrd+"ed")
        print(wrd+"ed")
        print(wrd+"ing")
else:                                                               #
noun handeling
    wrd = input("Enter the word: ")
    print("")
    print("Word forms:")
    if (wrd[-1] == "s" or (wrd[-1] == "h" and wrd[-2] == "c") or
(wrd[-1] == "h" and wrd[-2] == "s") or wrd[-1] == "x" or wrd[-1] == "z"
):         # branch, dish, Quiz
        print(wrd)
        print(wrd+"es")
    elif (wrd[-1] == "f"):
# calf, elf, half, leaf
        print(wrd)
        print(wrd[:-1]+"ves")
    elif (wrd[-1]=="e" and wrd[-2] == "f"):
# knife, life, wife
        print(wrd)
        print(wrd[:-2]+"ves")
    elif (wrd[-1]=="y" and (wrd[-2] == "a" or wrd[-2] == "e" or wrd[-2]
== "i" or wrd[-2] == "o" or wrd[-2] == "u")):
# annoy, toy, boy
```

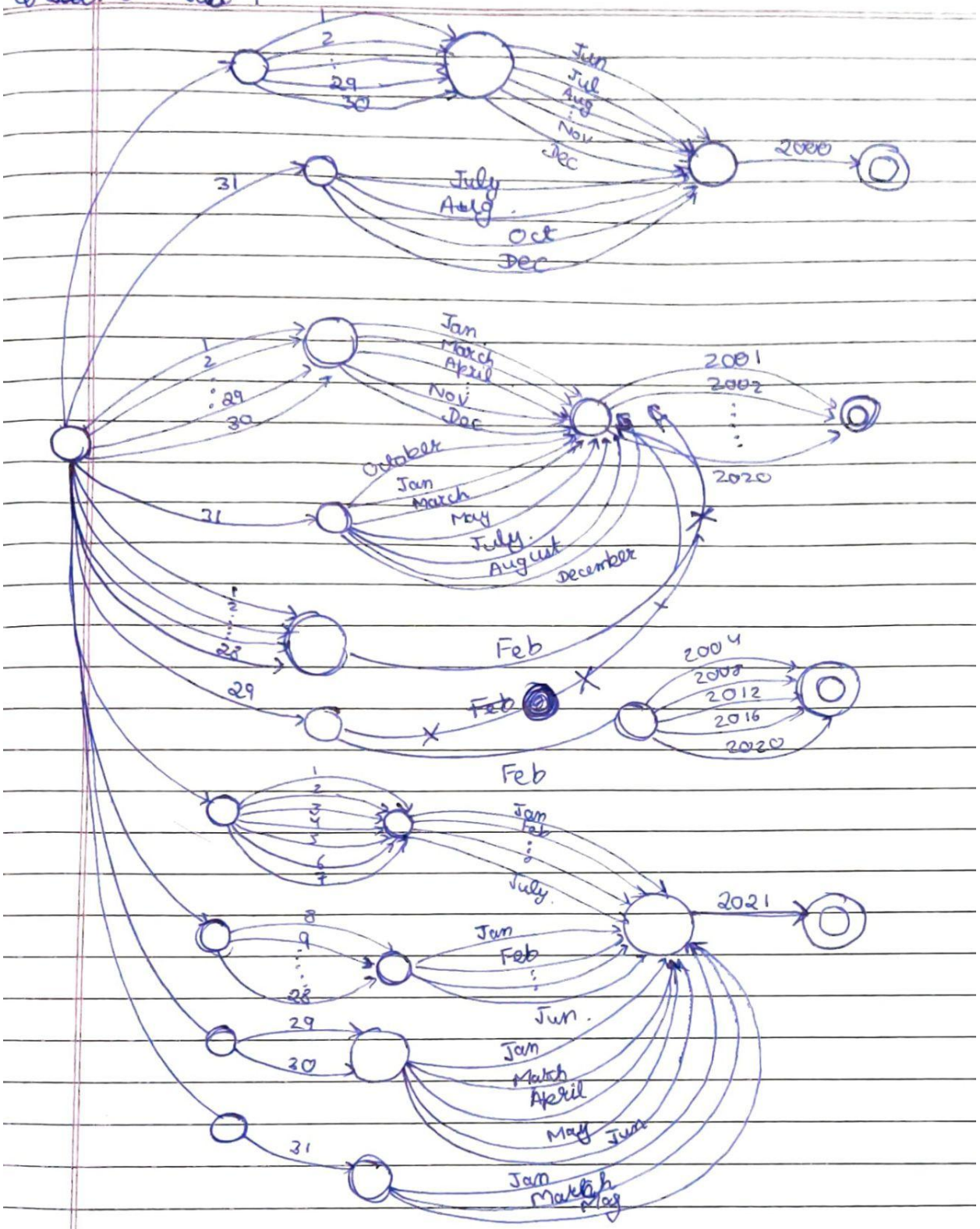
```
        print(wrd)
        print(wrd+"s")
    elif(wrd[-1] == "y"):
# baby, berry, bully, bunny
        print(wrd)
        print(wrd[:-1]+"ies")
    elif(wrd[-1]=="o"):
# hero, potato, tomato
        print(wrd)
        print(wrd+"es")
    elif(wrd[-1] == "n" and wrd[-2]== "o"):
# phenomenon, criterion
        print(wrd)
        print(wrd[:-2]+"a")
    else:
# cat, house
        print(wrd)
        print(wrd+"s")
```

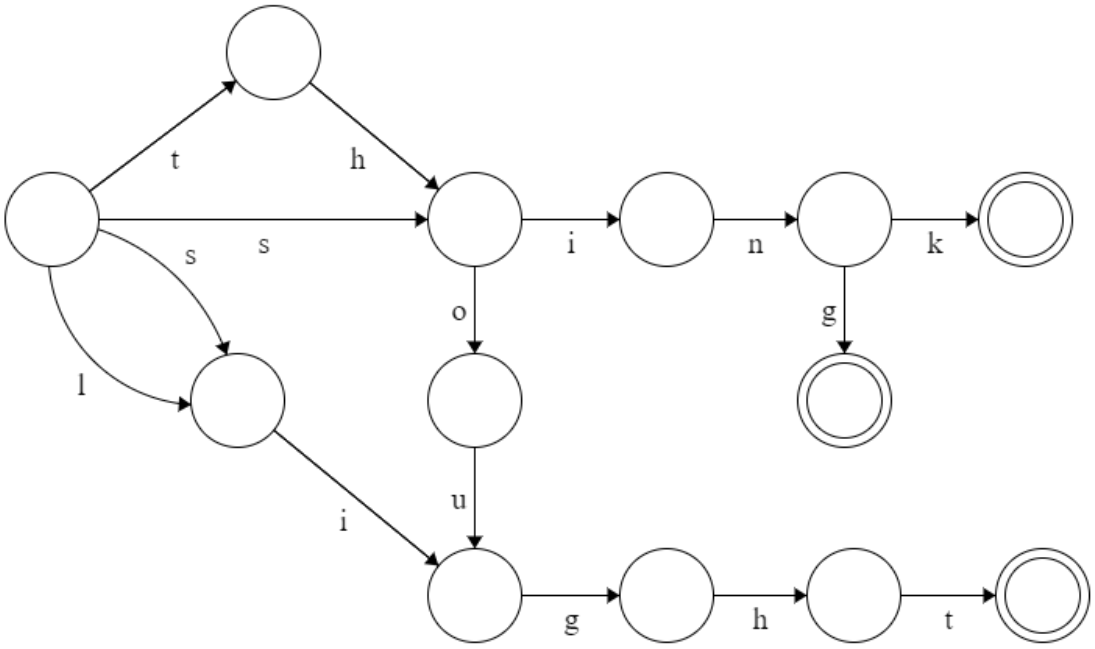
Question 2:

```
wrd = input("Enter the wrd: ")

   # returns first occurrence of Substring
a = wrd.find('acy')
# print(a)
b = wrd.find('dom')
# print(b)
c = wrd.find('er')
# print(c)
d = wrd.find('ism')
# print(d)
e = wrd.find('ment')
# print(e)
if d != -1:
        print("Root=", wrd.replace('ism', 'ist'), "Affix= -ism")
# communism, narcissism, skepticism
        quit()
elif b != -1:
        print("Root= ", wrd.replace('dom', ''), "Affix= -dom")       #
kingdom
        quit()
elif e != -1:
        print("Root= ", wrd.replace('ment', '', 1), "Affix= -ment")  #
endorsement
        quit()
elif a != -1:
        print("Root= ", wrd.replace('acy', 'ate'), "Affix= -acy")    #
privacy
        quit()

a1 = wrd.find('ship')
b1 = wrd.find('ied')
c1 = wrd.find('ing')
d1 = wrd.find('en')
e1 = wrd.find('ung')
f = wrd.find('al')
g = wrd.find('ate')
h = wrd.find('anti')
i = wrd.find('hyper')
```

```
if a1 != -1:
        print("Root= ", wrd.replace('ship', '', 1), "Affix= -ship")  #
fellowship
        quit()
elif h != -1:
        print("Root= ", wrd.replace('anti', '', 1), "Affix= anti-") #
anticlimax, antoseptic, antibody
        quit()
elif i != -1:
        print("Root= ", wrd.replace('hyper', '', 1), "Affix= hyper-") #
hyperactive, hypersensitivity, hypercritical
        quit()
elif b1 != -1:
        print("Root= ", wrd.replace('ied', 'y', 1), "Affix= -ied")   #
unified, verified
        quit()
elif g != -1:
        print("Root= ", wrd.replace('ate', 'ation'), "Affix= -ate")  #
regulate, eradicate, enunciate
        quit()
elif c != -1:
        print("Root= ", wrd.replace('er', ''), "Affix= -er")         #
trainer
        quit()
elif c1 != -1:
        print("Root= ", wrd.replace('ing', '', 1), "Affix= -ing")
        quit()
elif d1 != -1:
        print("Root= ", wrd.replace('en', '', 2), "Affix= -en")      #
enlighten
        quit()
elif e1 != -1:
        print("Root= ", wrd.replace('ung', '', 1), "Affix= -ung")
        quit()
elif f != -1:
        print("Root= ", wrd.replace('al', 'e'), "Affix= -al")        #
refusal
        quit()
```

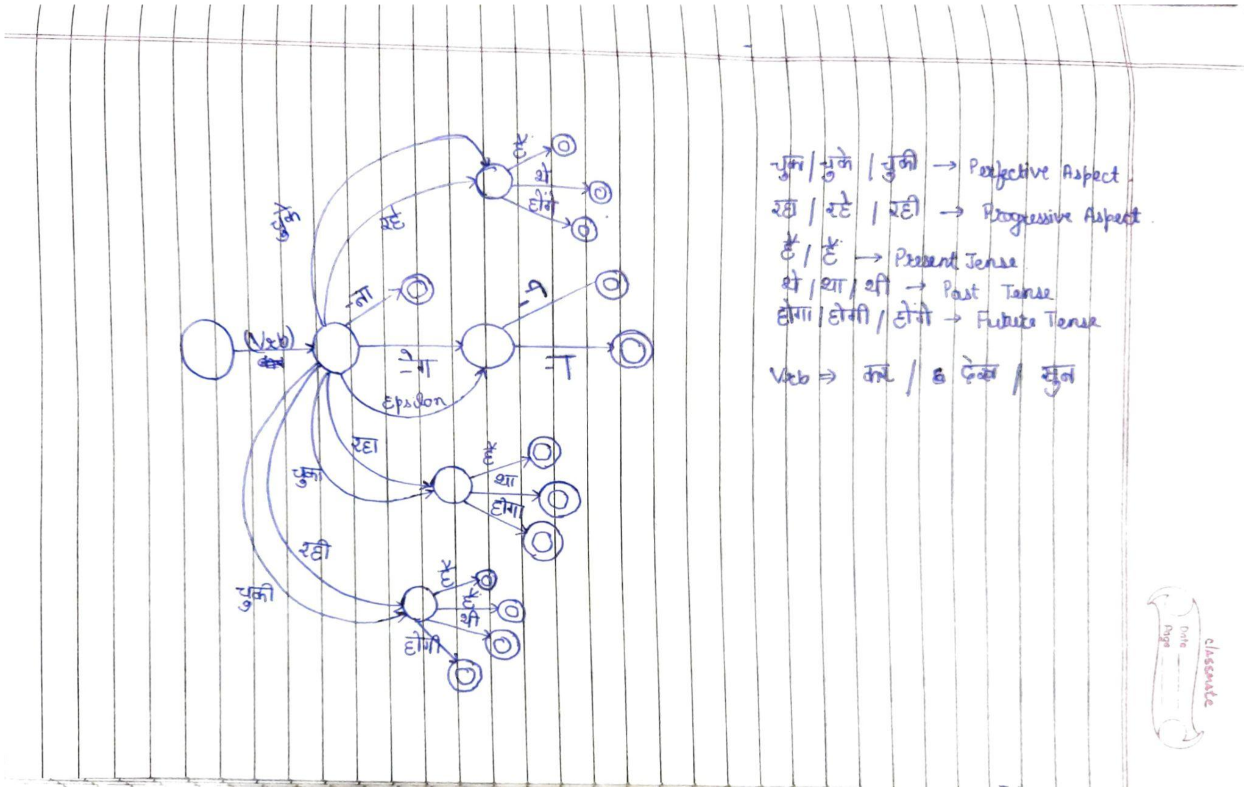
Q Part 3 → Part 1





Question 3 → part 2

Q4


चुका / चुके / चुकी → Perfective Aspect
रहा / रहे / रही → Progressive Aspect
है / हैं → Present Tense
थे / था / थी → Past Tense
होगा / होगी / होंगे → Future Tense
Verb ⇒ कर / देख / सुन