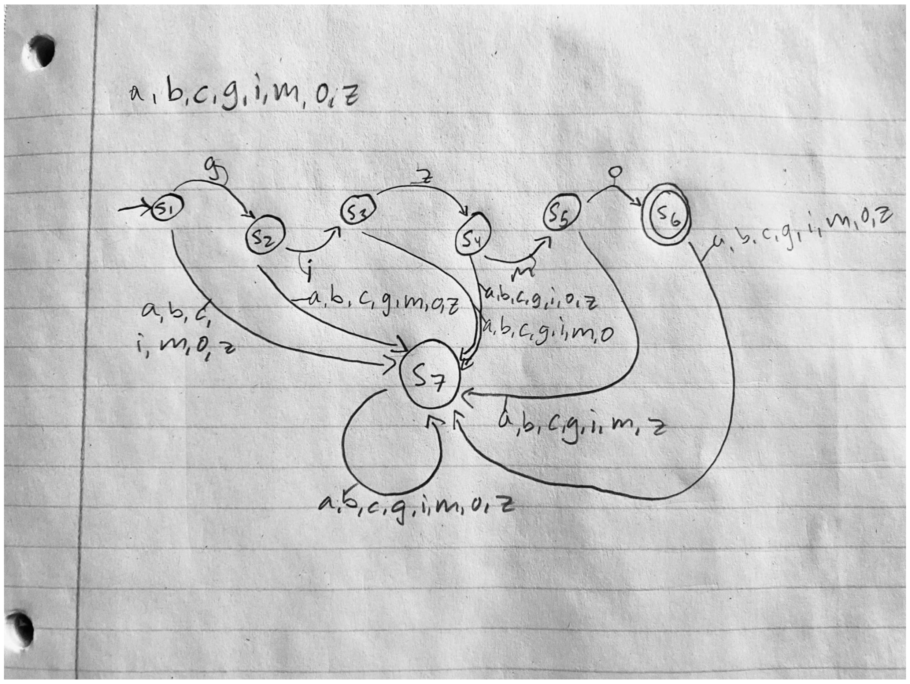
On my honor, I have not given or received any unauthorized assistance on this exam.

Problem 0:



Problem 1:

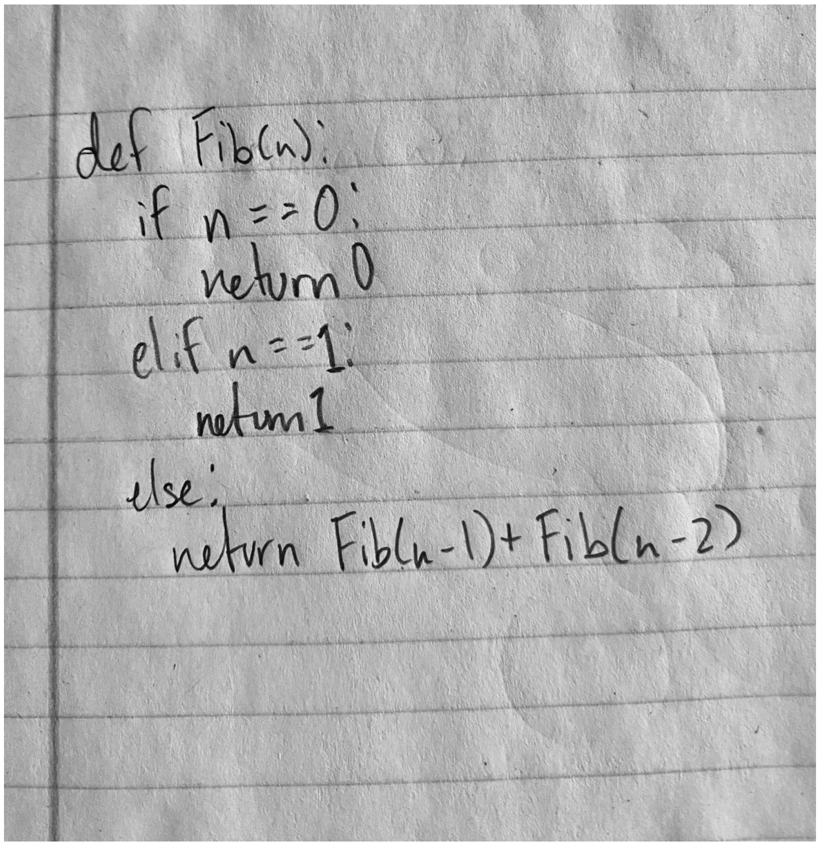
Type Name	Example
int	5
float	5.5
string	“5.5”
list	[5, 10, 20]
list	["gizmo", "sparky", "2001 Honda Odyssey"]
int	1110
string	"25"
int	25
float	25.0
string	"False"
boolean	True
list	[-1, 20, -3, 200,-1]

Problem 2:

- Fib(0) = 0
- Fib(1) = 1
- Fib(n) = Fib(n - 1) + Fib(n - 2)

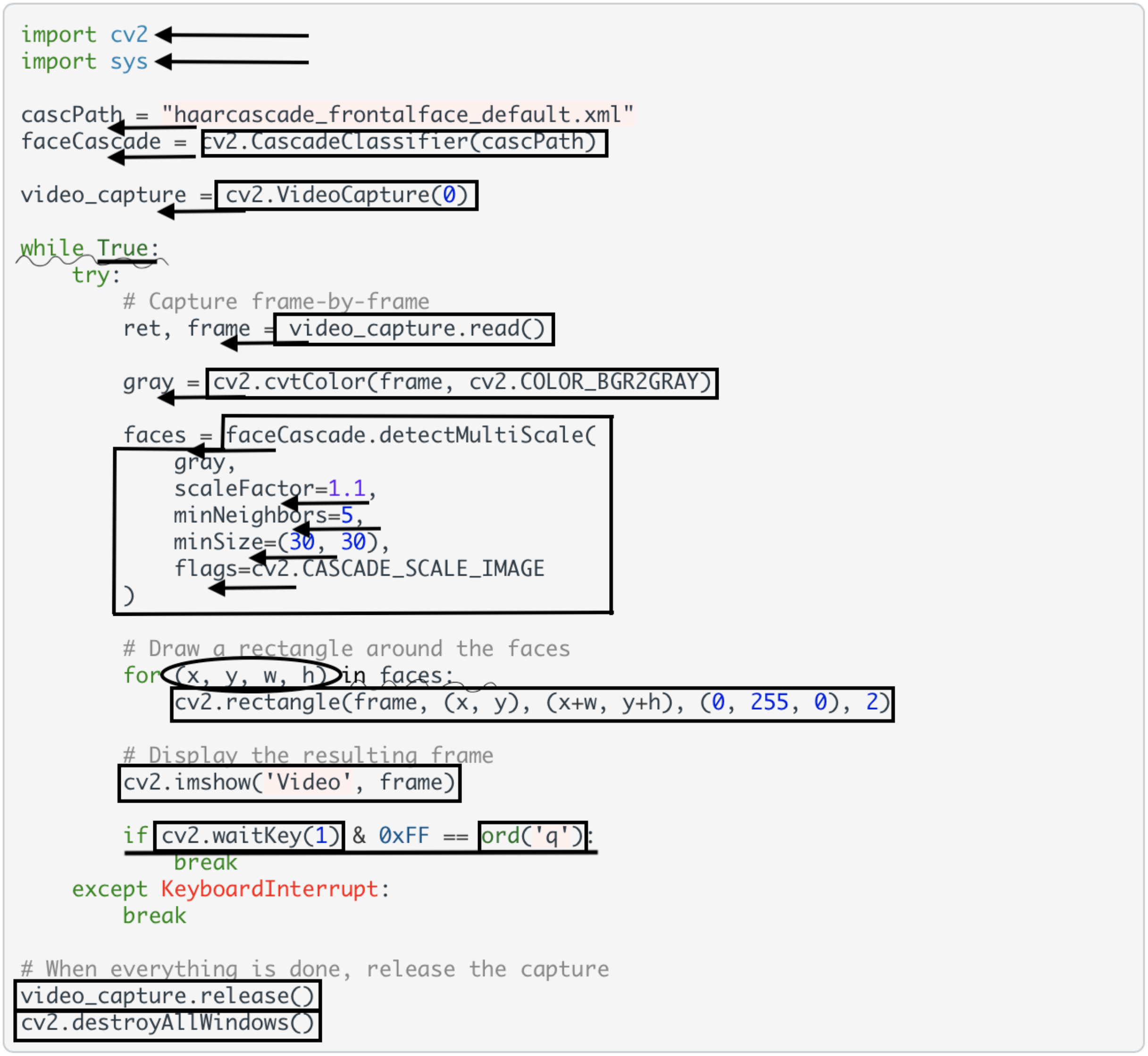
2a. The **first two** bullet points represent the **base** cases for the Fibonacci sequence, while the **third** bullet point represents the **recursive** case.

2b.



```
def Fib(n):
    if n==0:
        return 0
    elif n==1:
        return 1
    else:
        return Fib(n-1)+ Fib(n-2)
```

Problem 3:



```python
import cv2
import sys

cascPath = "haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(cascPath)

video_capture = cv2.VideoCapture(0)

while True:
    try:
        # Capture frame-by-frame
        ret, frame = video_capture.read()

        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

        faces = faceCascade.detectMultiScale(
            gray,
            scaleFactor=1.1,
            minNeighbors=5,
            minSize=(30, 30),
            flags=cv2.CASCADE_SCALE_IMAGE
        )

        # Draw a rectangle around the faces
        for (x, y, w, h) in faces:
            cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)

        # Display the resulting frame
        cv2.imshow('Video', frame)

        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    except KeyboardInterrupt:
        break

# When everything is done, release the capture
video_capture.release()
cv2.destroyAllWindows()
```

Problem 4:

4a. What kind of iteration would we use to ask for input? Why?

- A *while* loop would be best because the user will determine when the program ends and that cannot be accounted for beforehand.

4b. What kind of iteration would we use to reverse the string? Why?

- Depending on how the reverse occurs, either works, however, a *for* loop would most likely be used because the program can determine the length of the string.

4c.

- What built-in method would we use to tell how long each string is?
    - len(string_name)
- What built-in method could we use to get the list of integers [0 ... K - 1], where K is the length of the string?

    - input(“Give me a string”)
- Using these two methods, write the combined line of code that we would put to fill in the blank below:
    - `for i in len(input”Give me a string”):`
    - `      character = input_string[i]`

Problem 5:

5a. Give several (> 3) examples of variables that a Cat class would need to keep track of.

- *self.age = 0* for an initial age of the cats
- *self.name = name_input* to keep track of individual names of the cats
- *self.breed = breed_input* to keep track of individual breeds of cats
- *self.birthdate = birthday_input* to keep track of individual cats’ birthdays

5b. Give several (> 3) examples of methods that a Cat class would need to include to simulate cat-like activities like meowing.

- *def meow(self):* to meow, possibly print “meow”
- *def play_hunt(self):* to simulate playing, specifically “hunting”
- *def knead(self):* to perform the “make bread” motion that cats do
- *def birthday(self):* to “age up” a year

5c. For each of your examples in 5a and 5b, provide some justification for why a variable or a method is the correct way to handle each piece of Cat functionality.

- Age, name, breed, and birthdate are all pieces of information that might be needed to be accessed in other definitions, such as in a case of visiting the “vet” and requiring all the above information.
- Meow, play_hunt, knead, and birthday are specific actions that the cat may repeat several times throughout the program. They are also, in general, universal cat behavior.

Problem 6:

6a. What will happen when we run x + y ?

- An error would occur because we are attempting to add an int and string

6b. What will happen when we run str(x) + y ?

- Assuming it’s called as a print, it would print out 1616

6c. What will happen when we run x + int(y) ?

- Assuming it’s called as a print, it would print out 32

6d. What will happen when we run str(x + int(y)) ?

- Assuming it’s called as a print, it would print out 32

6e. What will happen when we run [x] + [y]?

- Assuming it’s called as a print, it would print out [16, “16”]