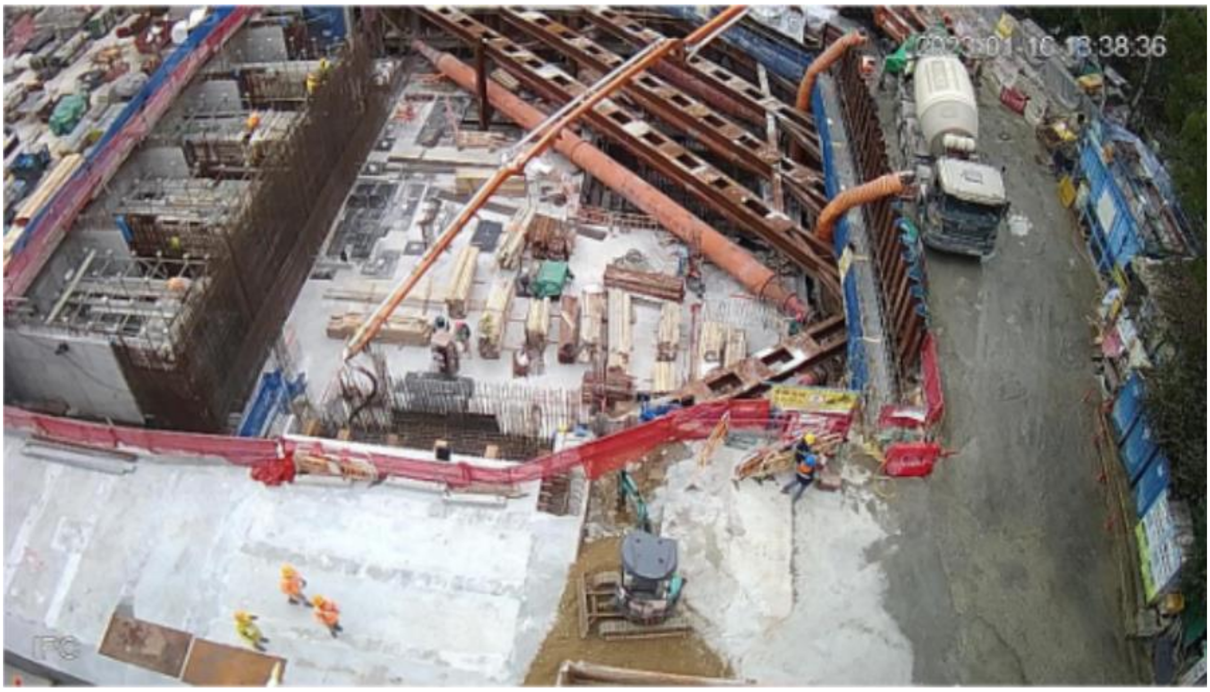
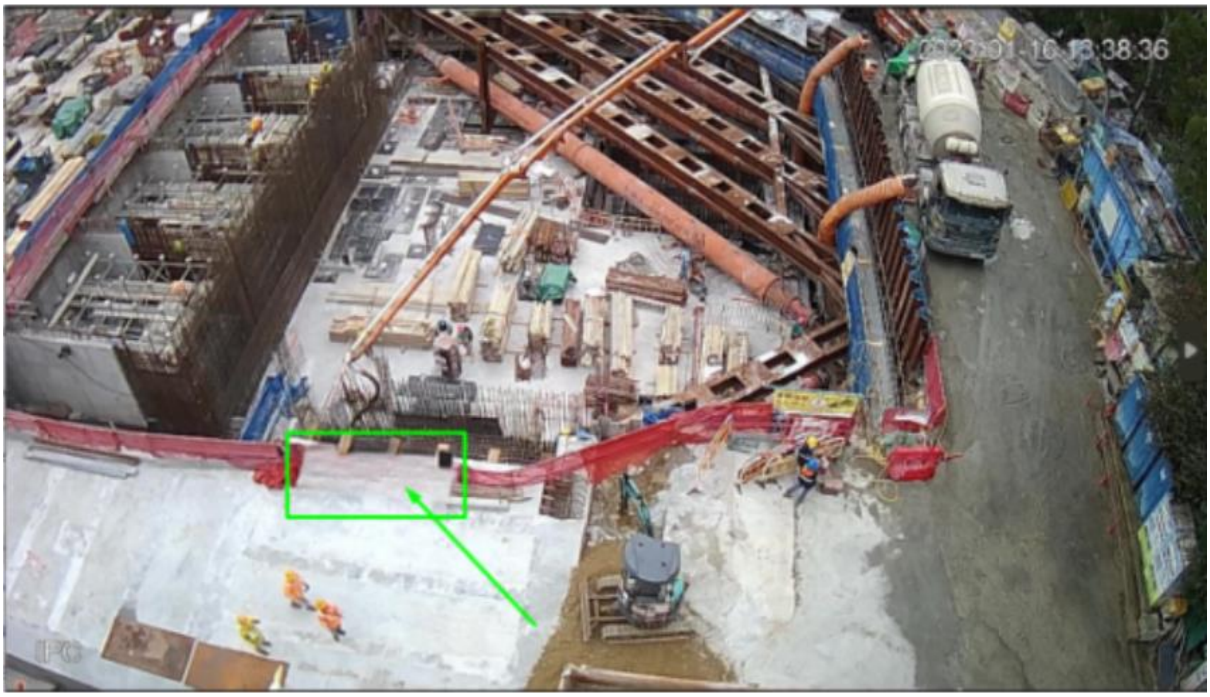
## Question 1. Open Edge

In construction, it is crucial to install structures on the edges to prevent workers on the roof from falling from the edges. Open edge means edges that are not guarded by some structures e.g. fence.

Examples are given in the images below:





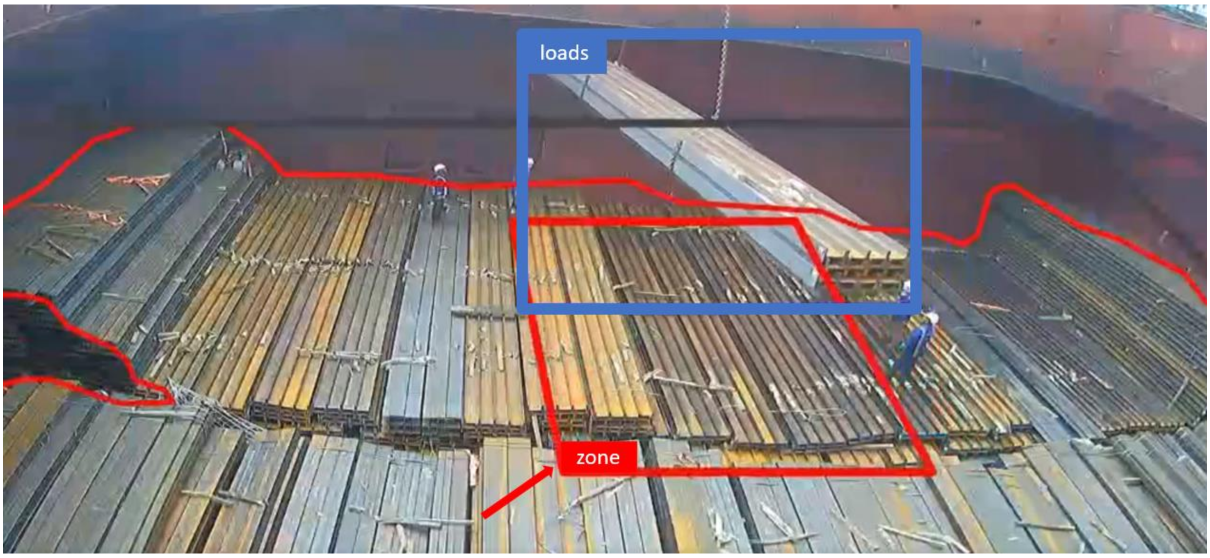
These structures take different forms. It may have different colours, shapes, orientations, and it may not be covered by a canva.

Please describe an approach that detects open edges like what is shown in the above bottom image, and it can be generalised to various camera angles, fence types or other scenarios.

## Question 2. Lifting Load Danger

In construction sites of tall buildings or skyscrapers, it's common to see heavy loads being lifted by cranes. To prevent suspended loads falling onto workers under lifting operations, it's necessary to locate the falling areas under the loads.

Images below are given for illustration:



The lifting loads can have different shapes, orientations, dangling locations.

Please describe an approach that detects/locates the falling zones of the suspended loads like what is shown in the above image, and it can be generalised to various camera angles, load types or other scenarios.

## Question 3. Class hierarchy

(a) Create a parser to read class hierarchy files and a data structure to efficiently (minimise time complexity) perform the following operations.

- Find all siblings class of a class name
- Find the parent class of a class name
- Find all ancestor classes of a class name
- Find if both class 1 and class 2 belong to the same ancestor class(es)

Definition:

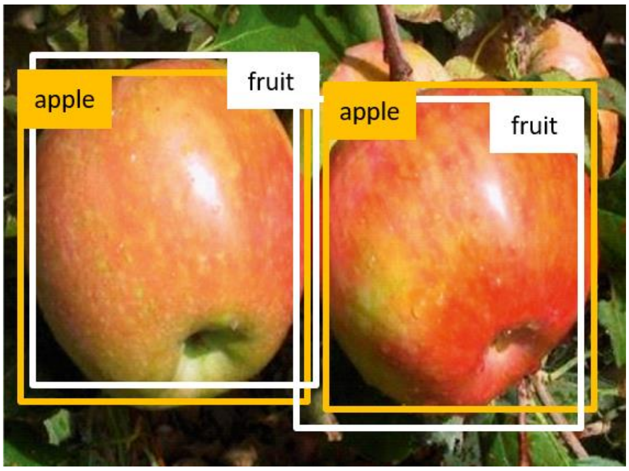
- Parent class: immediate superclass (up to 1 level in hierarchy)
- Ancestor class: all the superclasses (including parent of the parent class)
- Sibling class: classes that belong to the same parent class

You are provided with two files:

- hierarchy.txt: each line consists of a parent class and a child class
- id_to_name.txt: each line consists of a category id and its name

Please create a Github repository with public visibility containing your implementation and provide its link when you submit.

(b) It is not uncommon to see one object that is hierarchically labelled with parent and child classes. Softmax loss is a typical loss function in object detection training. But one drawback is that the scores of the labels belonging to one object suppresses each other.



Suppose all the labels including those from parent and children classes have to be used. Suggest some alternative loss functions to mitigate the above problem and elaborate the rationale behind their designs.

(c) Non maximum suppression is commonly used to remove duplicate bounding boxes with high IoU. Please modify the NMS algorithm to achieve the following:

1. Select boxes with confidence threshold that exceed a threshold
2. For highly overlapped boxes that belong to the same class, eliminate the one with lower confidence
3. For highly overlapped boxes that belong to different classes, eliminate the one that is the parent class of another box. For instance, box1 (Apple) and box2 (Fruit) are highly overlapped. Box 2 should be eliminated since box2 is a parent class of box1
4. The NMS function should accept arguments:
   i) bounding box proposals
   ii) scores/confidence
   iii) class
   iv) class hierarchy
   v) score threshold
   vi) iou threshold

Please use the same repository in (a) to submit your implementation for this question.