Problem 1

You are given an array of objects representing items to be put in a knapsack. The objects have 3 attributes: name, weight, and value. The items need to be selected so that the total weight does not exceed the maximum weight and the value is maximized.

knapsack([{ name:'map', weight:9, value:150 }, { name:'compass', weight:13, value:35 }, { name:'water', weight:153, value:200 }, { name:'sandwich', weight:50, value:160 }, { name:'glucose', weight:15, value:60 }, { name:'tin', weight:68, value:45 }, { name:'banana', weight:27, value:60 }, { name:'apple', weight:39, value:40 }], 100) should return 405.

knapsack([{ name:'map', weight:9, value:150 }, { name:'compass', weight:13, value:35 }, { name:'water', weight:153, value:200 }, { name:'sandwich', weight:50, value:160 }, { name:'glucose', weight:15, value:60 }, { name:'tin', weight:68, value:45 }, { name:'banana', weight:27, value:60 }, { name:'apple', weight:39, value:40 }], 200) should return 510.

knapsack([{ name:'cheese', weight:23, value:30 }, { name:'beer', weight:52, value:10 }, { name:'suntan cream', weight:11, value:70 }, { name:'camera', weight:32, value:30 }, { name:'T-shirt', weight:24, value:15 }, { name:'trousers', weight:48, value:10 }, { name:'umbrella', weight:73, value:40 }], 100) should return 145.

knapsack([{ name:'cheese', weight:23, value:30 }, { name:'beer', weight:52, value:10 }, { name:'suntan cream', weight:11, value:70 }, { name:'camera', weight:32, value:30 }, { name:'T-shirt', weight:24, value:15 }, { name:'trousers', weight:48, value:10 }, { name:'umbrella', weight:73, value:40 }], 200) should return 185.

knapsack([{ name:'waterproof trousers', weight:42, value:70 }, { name:'waterproof overclothes', weight:43, value:75 }, { name:'note-case', weight:22, value:80 }, { name:'sunglasses', weight:7, value:20 }, { name:'towel', weight:18, value:12 }, { name:'socks', weight:4, value:50 }, { name:'book', weight:30, value:10 }], 100) should return 237.

knapsack([{ name:'waterproof trousers', weight:42, value:70 }, { name:'waterproof overclothes', weight:43, value:75 }, { name:'note-case', weight:22, value:80 }, { name:'sunglasses', weight:7, value:20 }, { name:'towel', weight:18, value:12 }, { name:'socks', weight:4, value:50 }, { name:'book', weight:30, value:10 }], 200) should return 317.'

1st problem Answer:

```
def knapscak(n,W):
    if (n == 0 or W == 0):
        return 0
    elif (weights_list[n-1] > W):
        return knapscak(n-1,W)
    else:
        ans = knapscak(n-1,W)
        ans = max(ans, knapscak(n-1,W-weights_list[n-1])+ values_list[n-1] )
        return ans


def get_weight_and_values_list(dict_list):
    weights_list = []
    for dict_a in dict_list:
        weights_list.append(dict_a["weight"])
    weights_list.sort()
    values_list = []
    for i in range(len(weights_list)):
        weight_a = weights_list[i]
        for dict_a in dict_list:
            if weight_a == dict_a["weight"]:
                value = dict_a["value"]
```

```
            values_list.append(value)
    return weights_list,values_list


dict_list = [
{ "name":'waterproof trousers', "weight":42, "value":70 },
{ "name":'waterproof overclothes', "weight":43, "value":75 },
{ "name":'note-case', "weight":22, "value":80 },
{ "name":'sunglasses', "weight":7, "value":20 },
{ "name":'towel', "weight":18, "value":12 },
{ "name":'socks', "weight":4, "value":50 },
{ "name":'book', "weight":30, "value":10 }
]
weights_list,values_list = get_weight_and_values_list(dict_list)
n = len(weights_list)
W = 200
result = knapscak(n,W)
print(result)
```

Problem 2

Given a string (string brackets) containing just the characters '(', ')', '{', '}', '[' and ']', return a result to determine if the input string is valid. A valid string must adhere to the following rules:

=> Open brackets must be closed by the same type of brackets.

=> Open brackets must be closed in the correct order.

Example

( ) [ ] { } should return true

([ )] should return false

{ { [] ( ) }) should return false

{ [( )] } should return true

2nd Problem Answer:

```
def is_string_valid(string):
    list_a = []
    for char in string:
        if char in ['(', '{', '[']:
            list_a.append(char)
        elif char == ')' and list_a[-1] == '(':
            list_a.pop()
        elif char == '}' and list_a[-1] == '{':
            list_a.pop()
        elif char == ']' and list_a[-1] == '[':
            list_a.pop()
```

```
        else:
            return False
    return not list_a
string = input()
print(is_string_valid(string))
```

Problem 3

Given a short video, (use your own > 60 second video), use OpenCV to clip a 5 second clip from the 00:30 mark to the 00:35 mark and draw a red 100 x 100 pixel sized box in the middle of the video.