

Coded by: Naira Jezreel Carigo and Frederick Masangkay

```
class RiverCrossing:

    class People:
        def __init__(self, name, weight):
            self.name = name
            self.weight = weight

    def __init__(self, people, supply, boat_capacity):
        self.people = sorted(people) # Sort people for easier access to lowest/highest
        self.supply = supply
        self.boat_capacity = boat_capacity
        self.left_side = self.people + [self.supply]
        self.right_side = []
        self.boat = []

    def cross_river(self):
        self.right_side.extend(self.boat)
        self.boat.clear()

    def return_to_other_side(self, person):
        self.left_side.append(person)
        self.right_side.remove(person)

    def solve(self):
        while self.left_side:
            # Load the boat with as many people as possible
            while self.left_side and sum(self.boat) + self.left_side[0] <= self.boat_capacity:
                lowest_value = self.left_side[0] # Get the lowest value
                self.left_side.remove(lowest_value) # Remove from left side
                self.boat.append(lowest_value) # Add to the boat

            self.cross_river()


            if self.left_side:
                one_person = self.left_side[0] # Get the next lowest value
                self.left_side.remove(one_person) # Remove from left side
                self.boat.append(one_person)
                self.cross_river()
                self.return_to_other_side(one_person)

            if self.right_side:
                highest_value = max(self.right_side) # Get the highest value
                if highest_value + sum(self.boat) <= self.boat_capacity:
                    self.right_side.remove(highest_value)
                    self.boat.append(highest_value)
                self.cross_river()

        # Output the results
        print("Left side:", self.left_side)
        print("Right side:", self.right_side)

# Initial setup
people = [90, 80, 60, 40]
supply = 20
boat_capacity = 100

# Create an instance of RiverCrossing and solve the problem
river_crossing = RiverCrossing(people, supply, boat_capacity)
river_crossing.solve()
```

 File "<ipython-input-1-724d66ba7e57>", line 5
self.name = name
^
IndentationError: expected an indented block after function definition on line 4