

```

class ItemValue:

    def __init__(self, wt_, val_, ind_):

        self.wt = wt_

        self.val = val_

        self.ind = ind_

        self.cost = val_ // wt_

    def __lt__(self, other):

        return self.cost < other.cost

def fractionalKnapSack(wt, val, capacity):

    iVal = [ItemValue(wt[i], val[i], i) for i in range(len(wt))]

    iVal.sort(reverse=True)

    totalValue = 0

    for i in iVal:

        curWt = i.wt

        curVal = i.val

        if capacity - curWt >= 0:

            capacity -= curWt

            totalValue += curVal

        else:

            fraction = capacity / curWt

            totalValue += curVal * fraction

            capacity = int(capacity - (curWt * fraction))

            break

    return totalValue

if __name__ == "__main__":

    wt = [10, 60, 20, 40]

    val = [50, 40, 100, 150]

    capacity = 50

    maxValue = fractionalKnapSack(wt, val, capacity)

```

```
print("Maximum value in Knapsack =", maxValue)
```

OUTPUT:

Maximum value in Knapsack = 225.0