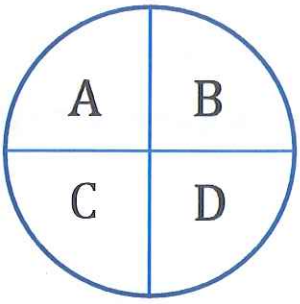
Question 1 – Integer Programming

8 marks

Space stations require occasional resupply of provisions that are delivered by rockets and, since rockets require their payloads to be balanced, deciding where to place cargo is critical to successful missions. The following figure shows a cross-section of the four compartments, A, B, C and D, in a typical delivery payload:



In order for the payload to be balanced, the total mass of cargo in section A must be equal to that of section D, and likewise for sections B and C. Note that section A does not need to equal section B and nor does section C need to be equal to section D.

There are 15 packages each of varying masses that must be included in the delivery. The masses of the packages in kilograms are as follows: 70, 90, 100, 110, 120, 130, 150, 180, 210, 220, 250, 280, 340, 350, 400. Each section must contain at least three packages and no compartment can hold more than 1,000 kg.

Formulate the problem of loading this delivery into the four compartments as an integer programming problem. Write the formulation in the space below. Implement your formulation in Python.

Sets P packages

S sections $= \{0, 1, 2, 3\}$ (A B C D)

Data m_p mass of package p

Variables $x_{ps} = 1$ if store package p in section s, 0 otherwise.

No objective

Constraints $A = D$: $\sum_{p \in P} m_p x_{p0} = \sum_{p \in P} m_p x_{p3}$

$B = C$ $\sum_{p \in P} m_p x_{p1} = \sum_{p \in P} m_p x_{p2}$

$\sum_{p \in P} m_p x_{ps} \leq 1000 \quad \forall s \in S$

$\sum_{p \in P} x_{ps} \geq 3 \quad \forall s \in S$

$\sum_{s \in S} x_{ps} = 1 \quad \forall p \in P$