**D. 市集採買**

　　終端雞市集，是一個以二元樹為原型，打造出的市集，強大的設計理念，吸引許多攤販進駐。所有攤販都被安排在除了葉節點以外的節點，而葉節點只會有「已經到盡頭」的告示牌。攤販們被規定只能在攤位上賣一件物品，並且需要標明這件物品的大小，以便大家放心採買。

烏骨雞和放山雞放學後，決定去終端雞市集採買，烏骨雞想要盡可能在不超過背包容量下，買到最多物品，但放山雞太懶惰，不想逛完整個市集，最後，兩雞達成協議，只會逛市集的其中一條，然後原路折返。

　　即便如此，烏骨雞還是想盡可能買到最多物品，因此牠想設計一個程式，輸入背包容量後，再輸入市集，就能獲得一條能買最多物品的路線。但因為可能同時存在多條路線，烏骨雞又有選擇困難，因此輸出永遠只會輸出最右邊那條。

　　而市集的輸入方法，以終端雞市集為例，為「」

**輸入格式**

　　第一行有一個不超過2147483647的正整數，第二行有2n個攤販所賣物品之大小，每個攤販所賣物品之大小為一個不超過2147483647的正整數。保證n不超過20且二元樹的深度不超過20層。

**輸出格式**

　　輸出只有一行，為一條能買最多物品且最右邊的路線。

**測試資料**

|  |  |
| --- | --- |
| 輸入範例1  1 7 1  10001010 | 輸出範例1  0 |
| 輸入範例2  1 7 0  10001010 | 輸出範例2  1 |
| 輸入範例3  5 3 1  10101011101010111010 | 輸出範例3  10101 |
| 輸入範例4  3 6 0  100101011011011111110 | 輸出範例4  110 |

**測試資料說明**

　　範例測資1、2請見題目敘述。

　　範例測資3中，「1010」的資料組為資料毀損的資料，輸出1，「1011」的資料組為驗證正確的資料，輸出0。

　　範例測資4中，「1001010」及「1101101」的資料組為資料毀損的資料，輸出1，「1111110」的資料組為驗證正確的資料，輸出0。

**配分**

|  |  |  |  |
| --- | --- | --- | --- |
| **記憶體限制** | 64MBytes | **評分方式** | Tolerant (寬鬆比對) |
| **編號** | **配分** | **時間限制** | **敘述** |
| #0~#1 | 6% | 0.1s | n=1,1≤m≤100 |
| #2~#4 | 9% | 0.1s | 1≤n≤100,1≤m≤100 |
| #5~#9 | 33% | 1s | 1≤n≤1000 |
| #10~#19 | 52% | 1s | 無特別限制 |