

Supermicro 題目說明

主題：

貨櫃堆疊空間利用之最佳化方案

說明：

隨著全球航運需求的增加，貨櫃運送價格等運輸成本在今年上半年即呈現飛快上漲的趨勢。如此快速的成本提升也牽動著全世界科技業與製造業的營運。在環境的變化與疫情衝擊下，我們需要善用資源以減少不必要的浪費。一個能讓貨櫃空間利用最佳化的解決方案不僅可以讓全球眾多產業節省資源，更能減少不必要的污染以達到永續發展的目的。

我們的目標是開發一個有效規劃貨櫃空間利用的軟體，由此軟體運算出的裝配方式能讓貨櫃內物件合理堆疊，不僅節省空間的利用，也考量到堆疊的結果是否穩固與重量分配。此軟體需能夠高效運算並以 3D 視覺化方式呈現規劃結果。

Supermicro 將提供的資源包含：

棧板、貨櫃等容器的規格與各種物件的裝箱尺寸、重量等等歷史資料以供運算。

題目定義：

給定多個大小不一容器，及多個大小不一的棧板，及多個大小不一的箱子。找出一個算法，使箱子能在一定限制下盡可能地堆上棧板，然後將這些堆好的棧板盡可能的塞進 container 裡。

題目架構拆分及比賽中的重點項目：

整個解決問題的步驟理論上如下：

[使用者輸入介面(定義不同物件的大小)] => [3d bin packing 算法] => [圖形渲染輸出]

前情提要，由於這次比賽會希望讓學生專注在 [3d bin packing 算法] 的部分，關於從使用者輸入介面的輸入接口需要目前不硬性要求。

supermicro 提供之渲染模組輸入格式（非強制使用）：

```
{
  "status": statusNumber,      //註解1
  "total_container_types": total_container_types, //註解2
  "total_pallet_types": total_pallet_types, //註解2
  "total_box_types": total_box_types, //註解2
  "containers": containers_array, //註解3
}
```

▼ 註解

▼ 註解 1

`status` 用來描述 3d bin packing 的結果狀態。

如：一個都裝不下，部分箱子能裝上棧板且棧板都能裝上貨櫃，或者匹配結束後尚有剩餘物件等等；或者嚴重出錯，一個太小的貨櫃導致根本沒東西能裝上去。

▼ 註解 2

`Total_${object_types}` 用來描述獨特的相同大小類型的物件集合，`${object}` 可為 `container`，`pallet` 或 `box`。

如：3*3*3 的 box 有 2 個、3*5*4 的 box 有 10 個，則 `total_box_types` 為 2。

▼ 註解3

`containers` 是一堆 `Container` 物件所組成的一維數列，用來描述貨櫃。

▼ `Container` 物件格式

```
Container {
  "ID": self.ID,          // 唯一id
  "TypeName": self.name,  // 類型總類
  "TypeIndex": self.type_index, // 該類型中的index
  "X": float(self.width),  // 寬
  "Y": float(self.height), // 長
  "Z": float(self.depth),  // 高
  "Weight_limit": float(self.max_weight), // 貨櫃重量限制
  "Fitted_items": FittedItemArray, // 裝上該貨櫃的棧板，詳見註解3-1
}
```

(`Weight_limit` 並非 `Weight_limit`)

▼ 註解3-1

`Container` 的 `Fitted_items` 其實是一堆 `pallet` 物件所組成的一維數列(array) 用來描述在這個 container 中的 pallet，除了 `RotationType`，該物件跟 `Container` 長得一樣。

```
{
  "ID": self.ID,          // 唯一id
  "TypeName": "122X102_0", // 棧板類別名稱
  "TypeIndex": 1,
  "X": 122.0,             // 寬
  "Y": 12.0,              // 長
  "Z": 102.0,             // 高
  "Weight": 23.0,
  "position_x": 0.0,
  "position_y": 0.0,
  "position_z": 0.0,
  "RotationType": 0,      // 棧板能在 container 裡做兩種角度的旋轉
  "Fitted_items": []
}
```

▼ 註解3-2

棧板的 `Fitted_items` 其實就是 boxes。

`Boxes` 物件如下：

```
{
  "ID": "057e2bb7-64cb-469a-a615-1c9c91aca474",
  "TypeName": "box1_1",
  "TypeIndex": 1,
  "X": 10.0,
  "Y": 10.0,
  "Z": 10.0,
  "Weight": 2.0,
  "position_x": 110.0,
  "position_y": 0.0,
  "position_z": 0.0,
  "RotationType": 0,      // box 能對 x,y,z 軸做旋轉，共有六種
}
```

▼ Position x, y, z 定義

`position_x`, `position_y`, `position_z` 是指物件在貨櫃或棧板上的位置。

軸的方向定義：

將右手舉起並舉直貼齊耳朵，指尖方向是 y 軸的方向，將右手向正前方舉起 90 度平行於地面，手掌朝下，指尖的方向是 z 軸，將右手朝你的右方舉起 90 度手掌平行於地面，這是 x 軸的方向。

物件的位置是指該物件的左下前角落位於 container 或者 pallet 的上位置，

物件的長寬高會從 x:0, y:0, z:0 朝各軸方向 "長" 出來。

如: 如果物件是 box, 我們要定義它在 pallet 上的位置，則在 8 個頂點中，找出 box 中頂點 (min(x), min(y), min(z))，

也就是左下前角落位於 pallet 上的位置，同時 pallet 的左下前角落被定義為 (0,0,0)。

如果物件是 pallet 則指它位於 container 裡的位置，過程同上。

▼ RotationType 定義

Pallet 是棧板，可以把它想像成是一張 a4 紙放在桌面上以各種方向旋轉 90 度的樣子。

實際上嚴謹一點的定義它會有 4 個旋轉方向 (對 y 軸做旋轉)。

但有兩個在實際操作上基本沒區別，所以我們只取兩個。

我們先將物件左下前角落放於 (0, 0, 0)。兩種旋轉分別為不旋轉，及朝順時鐘方向旋轉 90 度，然後再把左下角放回 (0, 0, 0)。

Table 1: Box rotation options

Rotation Type	First axis to rotate about	Second axis to rotate about
0	-	-
1	Z	-
2	Y	-
3	X	Y
4	X	-
5	X	Z

▼ TypeIndex 定義

假設一堆 containers (這裡的 containers 可替換為 pallet 或 box)，這一堆 containers 數量為10，其中 a 類型 3 個、b 類型 4 個、c 類型 3 個，所以共有 3 種類型。

我們可以建一個字典:

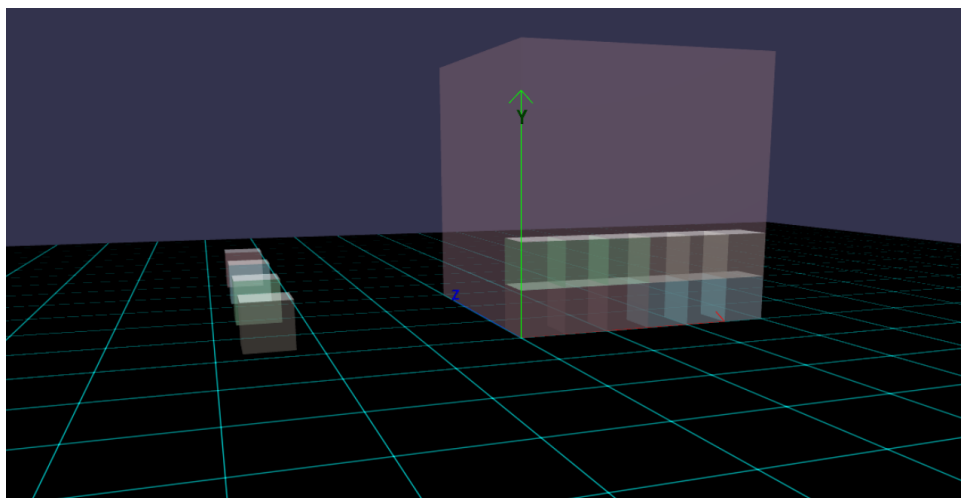
```
Int i = range(0, $total_type_numbers)
Dict[$typename: $i]
Dict["a"]=0 // 數字為 typeindex
Dict["b"]=1 // 數字為 typeindex
Dict["c"]=2 // 數字為 typeindex
```

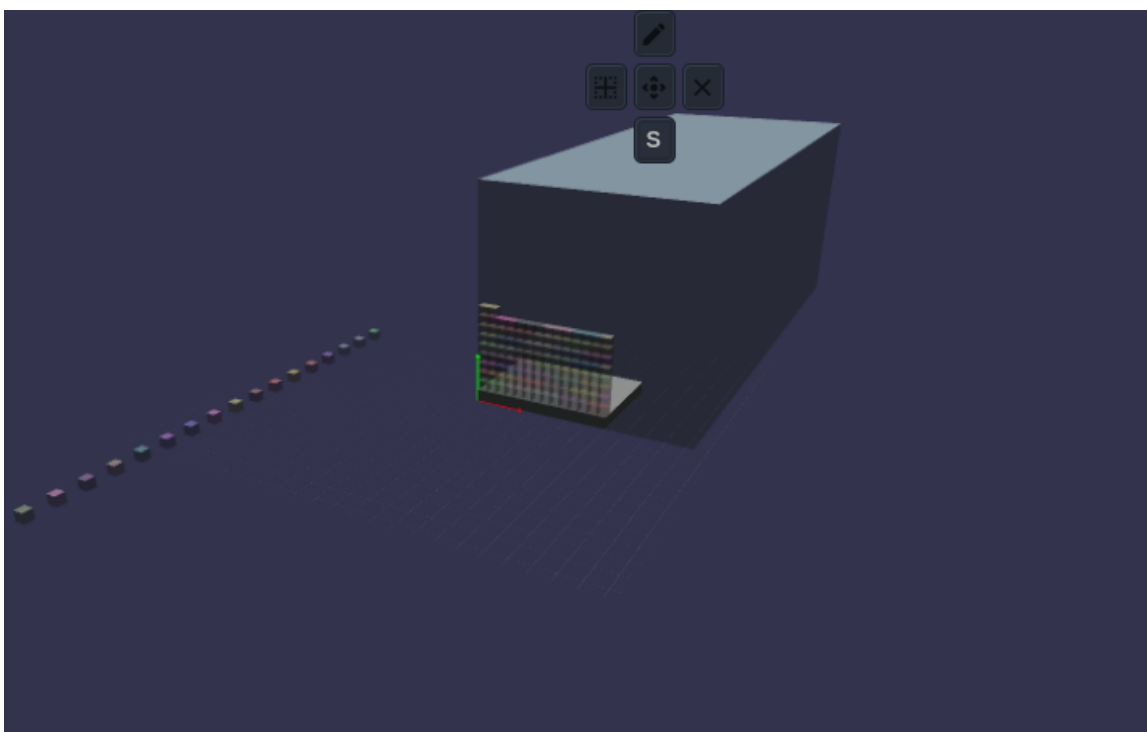
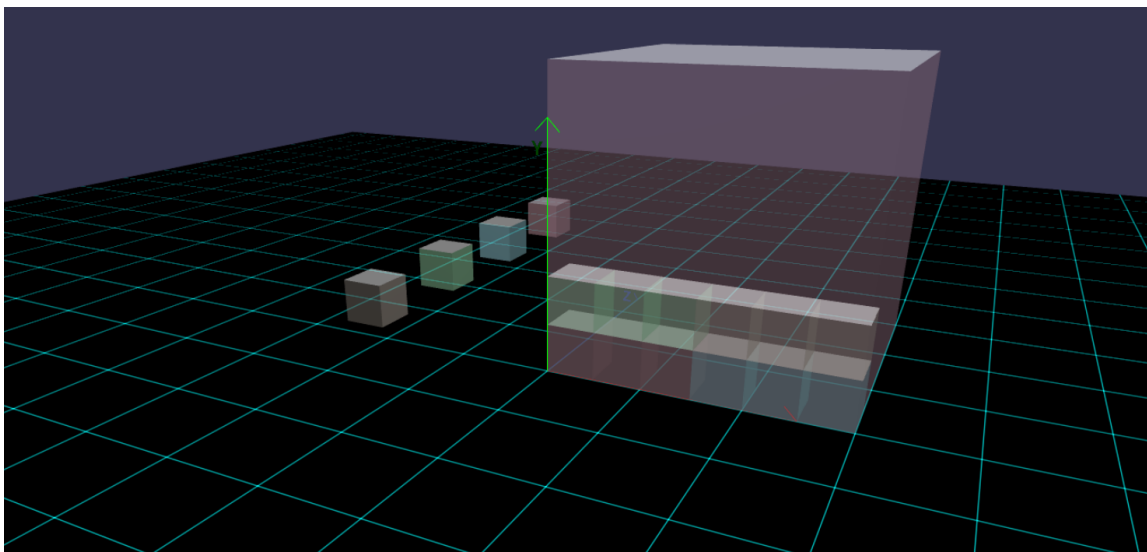
順序並不是重點，因為 TypeIndex 只是為了在渲染時能快速的從事先建好的 materials 中快速的選擇著色 (為了區分相同物件但不同類別的實體) 如上述例子，TypeIndex 不同能讓 a, b, c 為不同顏色。

但同樣是 a 類型的 container 將會是同一種顏色；這裡可能會有疑惑為何不用 typename 做區分就行，因為 typename 是使用者輸入的類型名稱，而並不能保證使用者是否會輸入多種 typename 相同，但長、寬、高不同的 object。

- 渲染模組 GitHub repo：[3D-Binpacking-GUI](#)
- 渲染模組輸入格式之範例請見下方「渲染模組輸入格式範例」

▼ 渲染圖





尺寸與限制條件：

我們打算將以 container 及 pallet 採用真實會用到的數據，但 box 的種類太多，出題時候乾脆隨機生成一批 boxes 來驗證算法的有效性。故以下僅附上 container 及 pallet 的數據及要求：

▼ 貨櫃尺寸

20'

Length 580CM

Width 227CM

Height 225CM

40'

Length 1180CM

Width 227CM

Height 225CM

▼ 標準棧板資訊(Unit: CM, KG)

Length	Width	Height	Weight	可否轉向
140	102	12	25KG	Y
122	102	12	22KG	Y
122	90	12	20KG	Y
110	75	12	12KG	Y
138	89	12	18KG	Y

▼ 棧板貨物堆疊與裝櫃原則

1. 單板貨物堆疊後毛重不超過450KG。
2. 相同料號優先堆疊於同一棧板。
3. 相近尺寸優先堆疊於同一棧版。
4. 大貨/重貨優先放置於棧板下層。
5. 盡量避免算法出現有物件懸空等不符合現實物理規律情況發生。

渲染模組輸入格式範例：

```
{
  "status": 130,
  "total_container_types": 1,
  "total_pallet_types": 2,
  "total_box_types": 5,
  "containers": [
    {
      "ID": "73bd0306-ccef-4e58-b9f3-7b73b263dc87",
      "TypeName": "helloContainer_0",
      "TypeIndex": 0,
      "X": 227.0,
      "Y": 225.0,
      "Z": 580.0,
      "Weight_limit": 20000.0,
      "Fitted_items": [
        {
          "ID": "b2c12820-e92e-4eb7-a38d-143b417f6112",
          "TypeName": "122X102_0",
          "X": 122.0,
          "Y": 12.0,
          "Z": 102.0,
          "Weight": 23.0,
          "position_x": 0.0,
          "position_y": 0.0,
          "position_z": 0.0,
          "RotationType": 0,
          "TypeIndex": 1,
          "Fitted_items": [
            {
              "ID": "3d62bf52-c021-4fab-b773-60f647e0363c",
              "TypeName": "test_1",
              "X": 100.0,
              "Y": 200.0,
              "Z": 100.0,
              "Weight": 1.0,
              "position_x": 0.0,
              "position_y": 0.0,
              "position_z": 0.0,
              "RotationType": 0,
              "TypeIndex": 0
            }
          ]
        }
      ]
    },
    {
      "ID": "702c9ef1-9563-4d14-b4b7-60f300e2a14b",
      "TypeName": "140X102_0",
      "X": 140.0,
      "Y": 12.0,
      "Z": 102.0,
      "Weight": 32.0,
```



```

"position_x": 122.0,
"position_y": 0.0,
"position_z": 0.0,
"RotationType": 3,
"TypeIndex": 1,
"Fitted_items": [
  {
    "ID": "00d2edf9-2fc8-4c25-a063-b608504270f4",
    "TypeName": "test_0",
    "X": 100.0,
    "Y": 200.0,
    "Z": 100.0,
    "Weight": 1.0,
    "position_x": 0.0,
    "position_y": 0.0,
    "position_z": 0.0,
    "RotationType": 0,
    "TypeIndex": 0
  },
  {
    "ID": "f939a6ec-721d-42a1-9c9d-92ec70f0b8e2",
    "TypeName": "box1_0",
    "X": 10.0,
    "Y": 10.0,
    "Z": 10.0,
    "Weight": 2.0,
    "position_x": 100.0,
    "position_y": 0.0,
    "position_z": 0.0,
    "RotationType": 0,
    "TypeIndex": 1
  },
  {
    "ID": "057e2bb7-64cb-469a-a615-1c9c91aca474",
    "TypeName": "box1_1",
    "X": 10.0,
    "Y": 10.0,
    "Z": 10.0,
    "Weight": 2.0,
    "position_x": 110.0,
    "position_y": 0.0,
    "position_z": 0.0,
    "RotationType": 0,
    "TypeIndex": 1
  },
  {
    "ID": "b1b3f58b-e086-4bed-bfa9-1f9f72ac0073",
    "TypeName": "box1_2",
    "X": 10.0,
    "Y": 10.0,
    "Z": 10.0,
    "Weight": 2.0,
    "position_x": 120.0,
    "position_y": 0.0,

```

```
        "position_z": 0.0,  
        "RotationType": 0,  
        "TypeIndex": 1  
    }  
]  
},  
    ],  
    "UnFitted_items": []  
}  
]
```