

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <unistd.h>
// #include<windows.h> //windows

using namespace std;

string u_name = "admin", u_password =
"admin", confirm_password = "";

int binSize = 0;

struct Item
{
    string name;
    int size;
};
vector<Item> items;
struct Bin
{
```

```
    int remainingCapacity;  
    vector<string> items;  
};
```

```
bool compare(const Bin &a, const Bin &b)  
{  
    return a.remainingCapacity <  
        b.remainingCapacity;  
}
```

```
void binPacking(int binSize, vector<Item>  
&items)  
{  
    vector<Bin> availableBins;  
    for (auto item : items)  
    {  
        bool placed = false;  
        for (auto &bin : availableBins)  
        {  
            if (bin.remainingCapacity >= item.size)  
            {  
                bin.remainingCapacity -= item.size;  
                bin.items.push_back(item.name);
```

```
        placed = true;
        break;
    }
}
if (!placed)
{
    availableBins.push_back({binSize -
item.size, {item.name}});
}
}
sort(availableBins.begin(),
availableBins.end(), compare);
for (int i = 0; i < availableBins.size(); i++)
{
    cout << "Bin " << i + 1 << ": ";
    for (int j = 0; j <
availableBins[i].items.size(); j++)
    {
        cout << availableBins[i].items[j] << " ";
    }
    cout << endl;
}
}
```

```

bool login()
{
    string name, password;
    int counter = 0;
    cout << " Enter the username:";
    cin >> name;
    cout << "Enter the password:";
    cin >> password;
    if (u_name == name && u_password ==
password)
    {
        return true;
    }
    else
    {
        cout << "Wrong Creditionals";
        cout << endl
            << "Enter the Creditonals Again\n";
        while (true)
        {
            if (counter = 3)

```

```
{
    break;
}
cout << " Enter the username:";
cin >> name;
cout << "Enter the password:";
cin >> password;
counter++;
if (counter == 1)
{
    cout << "You have left one chance to
login otherwise you will be blocked\n";
}
}
return false;
}
return false;
}
```

```
void takeSignUp()
```

```
{
    cout << "Enter your username:";
```

```
    cin >> u_name;
    cout << "Enter your password:";
    cin >> u_password;
    cout << "Enter confirm password:";
    cin >> confirm_password;
}

void signup()
{
    takeSignUp();
    int counter = 0;
    if (u_password != confirm_password)
    {
        cout << "Password and Confirm Password
doesn't match!\n";
        cout << "Retry" << endl;
        while (true)
        {
            if (counter == 3)
            {
                break;
            }
            if (counter == 1)
```

```

    {
        cout << "You have left one chance to
type write password otherwise you will have
to wait run the program again\n";
    }
    takeSignUp();
    counter++;
}
}
else
{
    cout << "Registered Successfully" << endl;
}
}

```

```

void chooseBinBox()

```

```

{
    const int Corrugated_Boxes = 1,
    Plastic_Boxes = 2, Rigid_Boxes = 3,
    Chipboard_boxes = 4;
    int choice;
    cout << "Choose Your Bin Box for your items
Packing from the list" << endl
    << endl;
}

```

```
    cout << "Press 1 for Corrugated_Boxes" << endl;
    cout << "Press 2 for Plastic_Boxes" << endl;
    cout << "Press 3 for Rigid_Boxes" << endl;
    cout << "Press 4 for Chipboard_boxes" << endl;
    cin >> choice;
    switch (choice)
    {
    case 1:
        cout << "Your Bin Box = Corrugated Box" << endl;
        binSize = 10;
        break;
    case 2:
        cout << "Your Bin Box = Plastic Box" << endl;
        binSize = 20;
        break;
    case 3:
        cout << "Your Bin Box = Rigid Box" << endl;
        binSize = 30;
        break;
```



case 4:

```
    cout << "Your Bin Box = Chipboard Box" << endl;
```

```
    binSize = 40;
```

```
    break;
```

```
default:
```

```
    break;
```

```
}
```

```
}
```

```
void addItem()
```

```
{
```

```
    cout << "Add Items" << endl;
```

```
    string productname;
```

```
    int weight;
```

```
    cout << "Enter the Product Name:";
```

```
    cin >> productname;
```

```
    cout << "Enter the Product Weight:";
```

```
    cin >> weight;
```

```
    items.push_back({productname, weight});
```

```
}
```

```
void showBin()
```

```

{
    binPacking(binSize, items);
}

int main()
{

    // vector<Item> items = {"item1", 5},
    {"item2", 4}, {"item3", 6}, {"item4", 3}, {"item5",
    3}, {"item6", 6}, {"item7", 22}};

    // binPacking(binSize, items);

    cout << "-----
    -----"

        << endl;
    cout << "\t\t\t"

        << "Bin Packing\n"

        << "-----
    -----"

        << "\n\n";

    // Sleep(milliseconds); windows
    unsigned int microsecond = 1000000; // for
    linux

        usleep(2 * microsecond);        // sleeps for 3
    second //for unix or linux

```

```
system("clear");
while (true)
{
    cout << "⚠ To Use This Application You  
must have to sign in" << endl;
    cout << endl
        << "Press 1 to Sign In" << endl
        << "Press 2 to Sign Up" << endl
        << "Press 3 to Exit" << endl;
    int option;
    cout << "Enter your choice:";
    cin >> option;
    if (option == 3)
    {
        break;
    }
    switch (option)
    {
    case 1:
        if (login())
        {
            system("clear");
```

```

        cout << "\n\t\t\t"
            << "Welcome to my Bin Packing
System made by Muhammed Husnain"
            << "\n\n";
        int ch;
        while (true)
        {
            cout << "Press 1 to choose your Bin
Box" << endl;
            cout << "Press 2 to add item" <<
endl;
            cout << "Press 3 to shows bins with
item names" << endl;
            cout << "Press 4 to exit" << endl;
            cin >> ch;
            if (ch == 4)
            {
                break;
            }
            switch (ch)
            {
            case 1:
                chooseBinBox();

```

```
        break;
    case 2:
        addItem();
        break;
    case 3:
        showBin();
        break;
    case 4:
        cout << "Exiting..." << endl;
        break;
    default:
        cout << "Invalid option" << endl;
        break;
    }
}

}

break;
case 2:
    signup();
    break;
case 3:
    break;
```

default:

```
    cout << "Invalid option" << endl;
```

```
    break;
```

```
    }
```

```
}
```

```
return 0;
```

```
}
```