Exercise 1:

Kamil Hassan’s:

#include <iostream>

using namespace std;

void printDistinct(int a[], int size){  
   int i, j;  
   for (i = 0; i < size; i++){  
      for (j = 0; j < i; j++){  
         if (a[i] == a[j]){  
            break;  
         }  
      }  
      if (i == j){  
         cout << a[i] << endl;  
      }  
   }  
}

int main()  
{  
    int arr [10];  
    int i = 0;  
    while (i != 10){  
        cin >> arr[i];  
        if(arr[i] >= 1 && arr[i] <= 15){  
            i++;  
        }  
        else {  
            cout << "Please input number between 1 - 15" << endl;  
        }  
    }

    printDistinct(arr, 10);

}

Choong Wei Jie’s:  
#include <iostream>

using namespace std;

int main()  
{  
    int num;  
    int nums[5];  
    int pointer = 1;  
    while (true)  
    {  
        cout << "Please enter a number: ";  
        cin >> num;  
        if (num > 15 || num < 0) {  
            cout << "Please enter a number within 1-15" << endl;  
            continue;  
        }

       bool status = true;  
        for (int i = 0; i <= pointer; i++)  
        {  
            if (nums[i] == num)  
            {  
                cout << "This number is used" << endl;  
                status = false;  
                break;  
            }  
        }  
        if (status)  
        {  
            cout << num << endl;  
            nums[pointer] = num;  
            pointer++;  
        }  
        if (pointer > 10) {  
            break;  
        }  
    }  
    return 0;  
}

Lee Kai Xian’s:

#include <iostream>

using namespace std;

int main()  
{  
  int arr[10]={1,3,5,7,9,11,13,15,1,3};  
  int n = sizeof(arr)/sizeof(arr[0]);

    int visited[n], count\_dis=0;

    for(int i=0; i<n; i++){

        if(visited[i]!=1){  
           int count = 1;  
           for(int j=i+1; j<n; j++){  
              if(arr[i]==arr[j]){  
                 count++;  
                 visited[j]=1;  
              }  
            }  
            if(count==1)  
             cout<<arr[i]<<" ";  
         }  
     }  
    return 0;  
}