```
#include <iostream>
#include <math.h>
#include <algorithm>
#include <cstdio>
#define S  18
#define N  6
using namespace std;
int calcs[N+1][S+1];
int num[] = {0,1,2,4,5,6};
int sizeo = sizeof(num)/sizeof(num[0]);
int numsize = sizeof(num)/sizeof(num[0]);
void printArray(){
        printf("       ");
        for(int i = 0;i<S+1;i++){
                printf("[%5d]",i);
        }
        cout<<endl;
        for(int i = 0;i<N+1;i++){
                for(int j = 0;j<S+1;j++){
                        if(j==0)
                                printf("[%5d]",num[i]);
                        printf("[%5d]",calcs[i][j]);
                }
                cout << endl;
        }
}
bool build(){
        for(int i = 0; i < sizeo;i++){
                for(int j = 0;j<18;j++){
                        if(i == 0 || j == 0){
                                calcs[i][j] = 0;
                        }else if(num[i] > j){
                                calcs[i][j] = calcs[i-
1][j];
                        }else{
                                int top = calcs[i-1][j];
                                int possible =num[i]+calcs[i-1][j-num[i]];
                                if(top == possible){
                                        cout << "Two unequeal subsets can equal :
"<<top<<endl;
                                        calcs[i][j] = max(top,possible);
                                        //return true;
                                }else{
                                        calcs[i][j] = max(top,possible);
                                }
                        }
                }
        }
        return false;
}

int main(){
        build();
        printArray();
        return 0;
}
```