1. (Randomized algorithms) Write a program to "discover" trigonometric identities. Your program should test all combinations of the trigonometric expressions shown below and use a randomized algorithm to detect the equalities. For your equality testing, generate random numbers in the $-\pi$ to $\pi$ range.

(a) $\sin(t)$  
(b) $\cos(t)$  
(c) $\tan(t)$  
(d) $\sec(t)$  
(e) $-\sin(t)$  
(f) $-\cos(t)$  
(g) $-\tan(t)$  
(h) $\sin(-t)$  
(i) $\cos(-t)$  
(j) $\tan(-t)$  
(k) $\frac{\sin(t)}{\cos(t)}$  
(l) $2\sin(t/2)\cos(t/2)$  
(m) $\sin^2(t)$  
(n) $1 - \cos^2(t)$  
(o) $\frac{1 - \cos(2t)}{2}$  
(p) $\frac{1}{\cos(t)}$

2. (Backtracking) The partition problem consists of determining if there is a way to partition a set of integers $S$ into two subsets $S_1$ and $S_2$ such that $\sum S_1 = \sum S_2$. Recall that $S_1$ and $S_2$ are a partition of $S$ if and only if $S_1 \cup S_2 = S$ and $S_1 \cap S_2 = \emptyset$. Write a function that solves the partition problem using backtracking. If a partition exists, your program should display it; otherwise it should indicate that no partition exists. For example, if $S = \{2, 4, 5, 9, 12\}$, your program should output the partition $S_1 = \{2, 5, 9\}$ and $S_2 = \{4, 12\}$ and if $S = \{2, 4, 5, 9, 13\}$ your program should indicate that no partition exists.

Given the little time available, a demo will not be required, thus it is very important that your report accurately describes your work.