**Rândul 1 (calculatoare impare)**

Salvaţi cele două proiecte în **c:\work** cu numele **r1pb1** pentru problema 1 respectiv r1pb2 pentru problema 2.

Pe prima linie a fiecărui cod sursă a programului C++, puneţi într-un comentariu prenumele vostru urmat de numele de familie.

1. Se citeşte un număr natural cu EXACT 4 cifre. Verificaţi dacă acest număr este palindrom. Veţi afişa pe ecran doar unul dintre mesajele "**este palindrom**" respectiv "**nu este palindrom**".

Un număr se numeşte palindrom dacă este egal cu oglinditul său (adică numărul citit de la dreapta la stânga). Spre exemplu numerele 4224 sau 1551 sunt palindromuri, pe când 1231 sau 1234 NU sunt. **(3p)**

2. Se citesc 6 numere reale, a, b, c, d, e, f. se garantează (deci NU mai este necesar să verificaţi acest lucru) că atât tripletul (a,b,c) cât şi tripletul (d,e,f) pot fi laturi ale unui triunghi.

Verificaţi dacă este vreo posibilitate ca, lipind (în plan) un triunghi de alt triunghi, să se obţină un patrulater. În caz afirmativ, calculaţi aria acestui patrulater. În caz negativ, afişaţi mesajul "Nu se poate obţine patrulater".

Indicaţie: pentru a putea obţine un patrulater, cel puţin o latură a primului triunghi trebuie să fie egală cu o latură a celui de-al doilea triunghi, prin urmare, verificaţi fiecare latură din primul triunghi cu fiecare latură din al doilea.

Ex: dacă citiţi valorile

3 4 5 12 5 13 răspunsul este "da", aria fiind egală cu 36.

Dacă citiţi valorile

3 4 5 12 6 13 răspunsul este "nu". **(5p)**