

Problema Nava Cosmică

E anul 2123 și sunteți angajați să scrieți softul pentru prima navă interstelară. Fiind un sistem extrem de complex, ați început prin implementarea sistemului de navigație. În particular, este nevoie de un program care să direcționeze nava și să țină minte distanța parcursă între planete. Ruta va fi calculată în felul următor: se caută cea mai apropiată planetă nevizitată și se parcurge distanța până acolo, până la vizitarea tuturor planetelor detectate. Dacă sunt mai multe planete la aceeași distanță față de poziția curentă, se alege prima din lista de la intrare.

Cerință

Scrieți un program care, primind la intrare coordonatele în spațiu ale planetelor și punctul de start, să calculeze distanța totală parcursă de navă pentru vizitarea tuturor planetelor.

Date de intrare

Se vor citi de la tastatură (fluxul *stdin*) pe prima linie un număr întreg **n** reprezentând numărul de planete. Pe următoarele **n** linii se află câte trei numere fracționare, reprezentând coordonatele pe x, y și z ale planetelor. Pe ultima linie se află poziția de start a navei, în același format ca și coordonatele planetelor. Coordonatele sunt date în parseci de la centrul galaxiei.

Date de ieșire

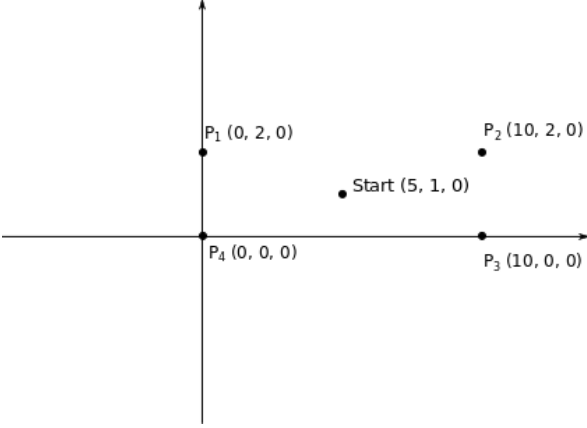
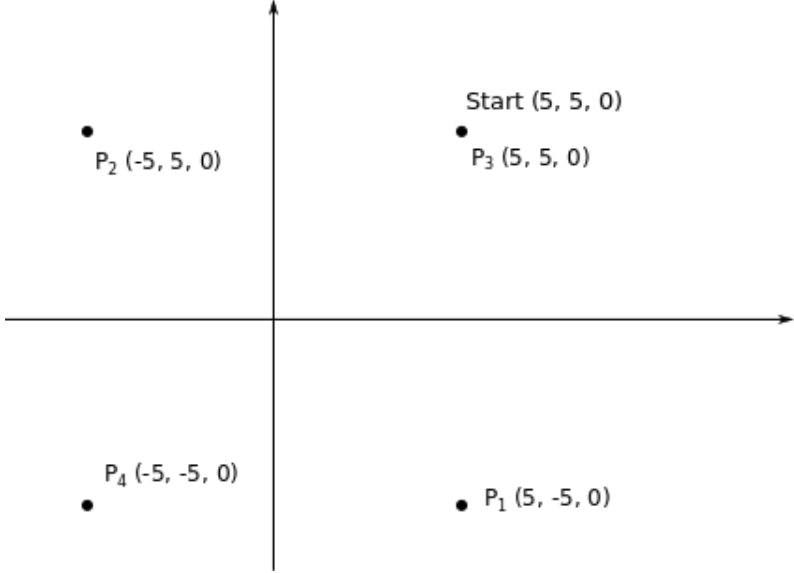
Programul va afișa pe ecran (stream-ul standard de ieșire) un singur număr fracționar, cu două zecimale (obținute prin rotunjire) reprezentând distanța totală parcursă de navă, în parseci, până la ultima planetă.

ATENȚIE la respectarea cerinței problemei: afișarea rezultatelor trebuie făcută EXACT în modul în care a fost indicat! Cu alte cuvinte, pe stream-ul standard de ieșire nu se va afișa nimic în plus față de cerința problemei; ca urmare a evaluării automate, orice caracter suplimentar afișat, sau o afișare diferită de cea indicată, duc la un rezultat eronat și prin urmare la obținerea calificativului „Respins”.

Restricții și precizări

1. $0 < n \leq 50000$
2. Se recomandă utilizarea tipurilor de date în virgulă mobilă în dublă precizie.
3. Calculele se vor face la precizie maximă, doar rezultatul final se va rotunji în scopul afișării.
4. Formula distanței între două puncte într-un spațiu 3-dimensional este similară cu formula pentru un spațiu 2D, care la rândul ei se poate deduce folosind teorema lui Pitagora.
5. **Atenție:** În funcție de limbajul de programare ales, fișierul ce conține codul trebuie să aibă una din extensiile .c, .cpp, .java, sau .m. Editorul web **nu va adăuga automat** aceste extensii și lipsa lor duce la imposibilitatea de compilare a programului!
6. **Atenție:** Fișierul sursă trebuie numit de candidat sub forma: <nume>.<ext> unde nume este numele de familie al candidatului și extensia este cea aleasă conform punctului anterior. Atenție la restricțiile impuse de limbajul Java legate de numele clasei și numele fișierului!

Exemple

Intrare	Ieșire	Explicație
4 0 2 0 10 2 0 10 0 0 0 0 0 5 1 0	19.10	 <p>Din punctul de start, toate planetele sunt la aceeași distanță, deci se va alege prima din listă, adică P1. De la Start la P1, distanța este de 5.099019514. Din punctul P1, cea mai apropiată planetă este P4. Distanța parcursă devine 7.099019514. Din P4, cea mai apropiată planetă nevizitată este P3, distanța parcursă devine 17.099019514. Apoi ultima deplasare este de la P3 la P2, distanța totală fiind 19.099019514. Afișat cu 2 zecimale, rezultatul este 19.10</p>
4 5 -5 0 -5 5 0 5 5 0 -5 -5 0 5 5 0	30.00	 <p>Punctul de start coincide cu P3 deci prima oprire are loc chiar acolo și distanța parcursă este 0. Cea mai apropiată planetă apoi, în ordinea de la intrare, este P1, distanța parcursă fiind de 10. De la P1 nava se va deplasa la P4 și apoi la P2. Distanța totală este 30.</p>

Timp de lucru: 120 de minute