

Termostat inteligent
Tema 2. POO
Ciudin Antonela 325CB

Cerinte:

1. Pentru stocarea datelor despre temperatura am creat o clasa "Room", cu variabilele "device_id", "surface" si doua ArrayList-uri care contin alte ArrayList-uri, unul dintre care cu temperaturi. Primul ArrayList contine 24 ArrayList-uri corespunzatoare fiecarui interval de timp, incepand cu pozitia 0 (ora 00:00) si terminand cu pozitia 23 (ora 23:00). Intervalul se calculeaza in metoda "calculateTimeInterval", unde se gaseste intervalul in care se incadreaza timestamp-ul primit. Pe fiecare pozitie se stocheaza temperaturile primite la comanda "OBSERVE" si dupa fiecare inserare se sorteaza crescator in metoda "sortTemp".
 - Asemator temperaturii, umiditatile sunt stocate intr-un ArrayList bidimensional similar, sortandu-le descrescator. Care se vor stoca numai in cazul in care initial este data umiditatea globala.
2. Pentru comanda "OBSERVE" stochez temperaturile pentru fiecare camera din ArrayList-ul "room_array", in 'matricele' de temperaturi declarate in clasa "Room". Pentru "TRIGGER HEAT" verific daca exista temperaturi, stocate in ultima ora (pe pozitia 23), apoi verific daca vectorul respectiv contine elemente si extrag primul element din vector, care fiind sortat crescator, pe prima pozitie sigur va contine valoarea minima. Le inserez intr-un ArrayList nou cu temperaturi minime.
 - Paralel verific daca exista umiditatea globala, extrag prima valoare, daca este, din ultimul interval de timp si le inserez in ArrayList cu maxime.

Dupa calculez media temperaturii (si a umiditatii daca este cazul), cu ajutorul metodei "calcAverage". Urmatorul pas este comanda de pornire sau nu a centralei: centrala NU se va porni in cazul in care temperatura medie este mai mica decat cea globala (cu o eroare de 0.05 permisa) sau in cazul in care umiditatea medie este mai mare decat umiditatea globala, in caz contrar centrala va fi pornita. Pentru comanda "TEMPERATURE", am actualizat temperatura globala corespunzator celei oferite in comanda. La comanda "LIST", initial apelez metoda "deleteRep" unde se elimina toate temperaturile care sunt indentice din acelasi interval de timp, apoi am folosit o variabila PrintWriter si printez in fisierul output temperaturile deja sortate