Solutie Ramen:

Se pot obține 32 de puncte simulând comenzile și mișcarea porțiilor pe bandă secundă cu secundă.

Încercând să obținem o soluție mai rapidă, observăm că procesarea comenzilor în ordinea timpilor nu este cea mai oportună: odată ce se primesc comenzi noi, acestea pot afecta orice altă comandă precedentă. Ne putem pune problema găsirii unei ordini de procesare care să aibă proprietatea că odată rezolvată o comandă, răspunsul pentru această comandă rămâne același până la finalul programului.

O ordonare care se pretează natural la această cerință este sortarea în ordine crescătoare după poziția clienților. Odată ce am aflat răspunsurile pentru acele NR comenzi care implică clienții cei mai apropiați de bucătărie, putem afla răspunsul și pentru a (NR + 1)-a comandă iar acesta va fi permanent, fiindcă în general o anumită comandă poate fi afectată doar de comenzi făcute de pe poziții mai mici.

Să vedem deci cum determinăm răspunsul pentru comanda de pozitie minima. Dacă această comandă este definită de valorile TT, PP, atunci ea ar putea fi satisfăcută cu orice comanda care are T(i) + D + PP >= TT (orice portie cu T(i) mai mic ar fi trecut deja de scaunul tinta inainte ca acesta sa fi facut comanda). Dintre acestea, raspunsul va fi dat de comanda cu T(i) minim. Cu alte cuvinte avem nevoie sa gasim cel mai mic timp T(i) cu proprietatea ca:

T(i) >= TT - D - PP

Raspunsul va fi apoi T(i) + D + PP. Apoi va trebui sa stergem aceasta comanda din multimea comenzilor care mai pot constitui raspunsuri. Solutiile care cauta si sterg aceste valori in timp liniar pot obtine 57 de puncte. Pentru o solutie mai rapida este necesar ca aceste operatii sa se efectueze in timp logaritmic. Acest lucru se poate realiza cu un arbore de intervale sau cu container-ul STL std::set<>.

O alta solutie, intr-un sens duala celei precedente, presupune crearea a doua tipuri de evenimente: aparitia unei comenzi pe banda, respectiv inceperea disponibilitatii unui client de a lua orice portie din fata sa. Aceste evenimente vor fi apoi parcurse in ordine crescatoare a timpilor, iar pentru fiecare comanda se va afla la ce client va ajunge. In general, o comanda va ajunge la clientul cu indice minim dintre cei disponibili la momentul respectiv. Apoi, clientul respectiv trebuie sters din multimea clientilor disponibili. Solutiile care cauta acest minim si il sterg in timp liniar pot obtine de-asemenea 57 de puncte. O solutie mai rapida se poate obtine efectuand aceste operatii cu ajutorul unui heap sau cu container-ul STL std::set<>.