

Method assign Pairs {

for EingetragenePerson i = 0 do

Person = EingetragenePerson[i]

if Person.Verfügbar do

for EingetragenePerson j = 0 do

ComparingPerson := EingetragenePerson[j]

Vergleiche die Kriterien der Personen miteinander

↳ if (Criteria.isOkay) then ordne zu potenziellen Pärchen.

for potenzielle Pärchen do

↳ Suche nach dem besten Pärchen

↳ bilde Paar

↳ Setze bezüglich der jeweiligen Person die Koordinate für Küche fest

Setze die Preferenzen

Setze Verfügbarkeit der jew. Person false

if (Person.übrig) then füg Warteschlange hinzu.

}

Erläuterung: Die Methode kriegt die Liste mit den Anmeldungen der Personen als Input daran hin wird über die Liste iteriert, und eine bestimmte Person[i] festgelegt. Daraufhin wird nochmals durch die Liste iteriert und eine Variable (ComparingPerson) ausgewählt. Dann werden die Kriterien wie Essensvorlieben oder Kitchenvorlieben abgecheckt. Wenn die Kriterien erfüllt werden, wird die jeweilige Comparing Person zu den potenziellen Pärchen (Paar) Liste hinzugefügt. Durch eine Iteration durch die Liste Potenzielle Pärchen wird anhand einer Iterativen Abfrage die beste geeignete Person festgestellt. Danach wird das Pärchen zusammen gefügt. Die Koordinaten der Küche werden bezüglich der Person koordiniert, welche eine Küche besitzt. Falls es Personen gibt, welche gar keinen potenziellen Pärchenpartner gefunden haben, kommen in die Warteschlange, wo sie dann später aufgerufen werden.

Create Groups (dist <> Pairs, boolean, Anzahl){

List PairsMaxSize;

maxPairsize := MaximalPersonAmount / 2

for Pairs i = 0 do

if (i > maxPairsize) then break.

↳ pairsMaxSize . add (pairs . get(i))

Warteschlange . add . (All Restpairs)

for pairsMaxSize , i = 0 do

↳ FoodPreference . add . (pairs . MaxSize . get(i)) // addet zum jcw. Foodpreference.

AllPreference := listOf(AllFoodPreferences)

allGroups := Die Anzahl der Pairs bis die Anzahl „Vielfache von 9 gefüllt“

Warteschlange . add (Rest der Vielfache).

Unterteile die Pärchen von allGroups zu den Speisen : {Vor, Haupt, Nach}

Bilde durch Verhettke Iteration die Gruppen.

}.

Erläuterung 8 Als Input , kriegt man eine Liste von Pairs, ein boolean und die Maximalanzahl an Personen.

In der ersten Iteration , wird die Anzahl der Pärchen , der Maximal - Anzahl der Personen angepasst. Der Rest , welche die Anzahl überschreiten würden kommen in die Warteschlange. In der nächsten Iteration werden die Pärchen in die jeweilige Liste ihrer Foodpreference unterteilt, um besser im Verlauf des Codes eingeordnet zu werden.

Wichtig für die erfolgreiche Gruppenbildung ist , dass die Anzahl der Pärchen eine Vielfache von 9 sind , um sie passend zu unterteilen. Dafür wurde eine Hilfsfunktion benutzt. Der Rest von der Vielfache kommt wieder in die Warteschlange Im Verlauf des Codes werden dann die Gruppen . Mithilfe der „optimierten“ Listen gebildet.