# C++ OOP Retake Exam – 27 April 2024

Submit your solutions here: <https://judge.softuni.org/Contests/Practice/Index/4741>

# Conference Participants

You have a conference participant management system, which uses the following classes to do its job:

|  |  |
| --- | --- |
| Participant | **The base class of all participants.** |
| ├── Organizer | **Base class for all organizers** |
| │ ├── OrganizerEmployee | **Employee of the organizer** |
| │ └── OrganizerManager | **Manager of the organizer** |
| ├── Visitor | **Base class for all visitors** |
| │ ├── AdultVisitor | **Adult (above 18 years’ old) participant** |
| │ └── MinorVisitor | **Minor (below 18 years’ old) participant** |
| └── Sponsor | **Base class for an employee of a sponsor** |
| ├── BronzeEmployee | **Employee from a bronze sponsor** |
| ├── SilverEmployee | **Employee from a silver sponsor** |
| └── GoldEmployee | **Employee from a gold sponsor** |

The system reads participant data, depending on each participant type, then builds the hierarchy of participants. For each different class we've got the different input and output. The base classes do not have input, as they cannot exist in the hierarchy.

The input data is classified in the following way for each separate class:

* Each participant data starts with the Type as follows: **OE, OM, AV, MV, SE** and **GE** (see below for identification which **Type** to which class refers)
* The Id of the participant, an int;
* The rest of the fields are only alphanumeric, e.g. cannot contain digits, spaces, or punctuation.
  + The only exception is the **Age** field of the **MinorVisitor**, which is an int like the Id.
* It's guaranteed that the Name + Surname pairs are unique
* It's guaranteed that each **MinorVisitor** participant will have a corresponding accompanying adult (e.g., there will be an adult with that pair of names)
  + There's no such guarantee for the Employee <-> Manager connection, and it does not matter from this application’s perspective.
* The data will be correct both syntactically and semantically.
* The input ends with **end**.

Here’s the input format of each class:

|  |  |
| --- | --- |
| Participant | ***Does not have specific input*** |
| ├── Organizer | ***Does not have specific input*** |
| │ ├── OrganizerEmployee | OE Id Name Surname ManagerName ManagerSurname  **Example:**  OE 2 Ivan Popov Stefan Petrov |
| │ └── OrganizerManager | OM Id Name Surname Department  **Example:**  OM 4 Stefan Petrov Sales |
| ├── Visitor | ***Does not have specific input*** |
| │ ├── AdultVisitor | AV Id Name Surname accountId  **Example:**  AV 6 Doncho Angelov donangel |
| │ └── MinorVisitor | MVId **Name Surname AdultName AdultSurname Age**  **Example:**  MV 8 Kalina Angelova Doncho Angelov 11 |
| └── Sponsor | ***Does not have specific input*** |
| ├── SilverEmployee | **SE** Id **Name Surname Company** accountId  **Example:**  SE 10 Stilyan Petrov CompanA stipe |
| └── GoldEmployee | **GE** Id **Name Surname Company TshirtSize**  **Example:**  GE 11 Marko Markov markma XXXL |

The system will read and classify each input line, and then will print out a participation summary and a list of all participants by type. See the Example section for details.

Your task is to study the class hierarchy and complete it, and then provide the needed operators, so that the output is successful. Your ZIP file should should contain the following files: **Solution.h**, **SilverEmployee.h**, and **Sponsor.h**.

## Example 1

|  |
| --- |
| ****Input**** |
| OM 1 Ivan Tenev Sales  **OE 2 Stanoika Ivanova Ivan Tenev**  **AV 23 Petar Petrov pepet**  **AV 21 Ivan Georgiev ivge**  **GE 8 Golden Aishe Fatctory XXXL**  **end** |
| ****Output**** |
| **Participants: 5, Organizers: 2, Sponsors: 1, Visitors: 2**  **Organizers:**  **- Tenev, Ivan | Sales | manager**  **- Ivanova, Stanoika | manager: Tenev, Ivan**  **Visitors:**  **- Petrov, Petar | pepet**  **- Georgiev, Ivan | ivge**  **Sponsors:**  **- Gold: Aishe, Golden | Fatctory | XXXL** |

## Example 2

|  |
| --- |
| ****Input**** |
| SE 10 Stilyan Petrov CompanA stipe  GE 11 Marko Markov markma XXXL  MV 8 Kalina Angelova Doncho Angelov 11  AV 6 Doncho Angelov donangel  OM 4 Stefan Petrov Sales  OE 2 Ivan Popov Stefan Petrov  **end** |
| ****Output**** |
| **Participants: 6, Organizers: 2, Sponsors: 2, Visitors: 2**  **Organizers:**  **- Petrov, Stefan | Sales**  **- Popov, Ivan | manager: Petrov, Stefan**  **Visitors:**  **- Minor: Angelova, Kalina | 11 | accompanied by: Angelov, Doncho**  **- Angelov, Doncho | donangel**  **Sponsors:**  **- Silver: Petrov, Stilyan | CompanA | stipe**  **- Gold:** Markov, Marko | markma | XXXL |

## Example 3

|  |
| --- |
| ****Input**** |
| AV 6 Doncho Angelov donangel  OM 4 Stefan Petrov Sales  **end** |
| ****Output**** |
| **Participants: 2, Organizers: 1, Sponsors: 0, Visitors: 1**  **Organizers:**  **- Petrov, Stefan | Sales**  **Visitors:**  **- Angelov, Doncho | donangel**  **Sponsors:**  **- none** |