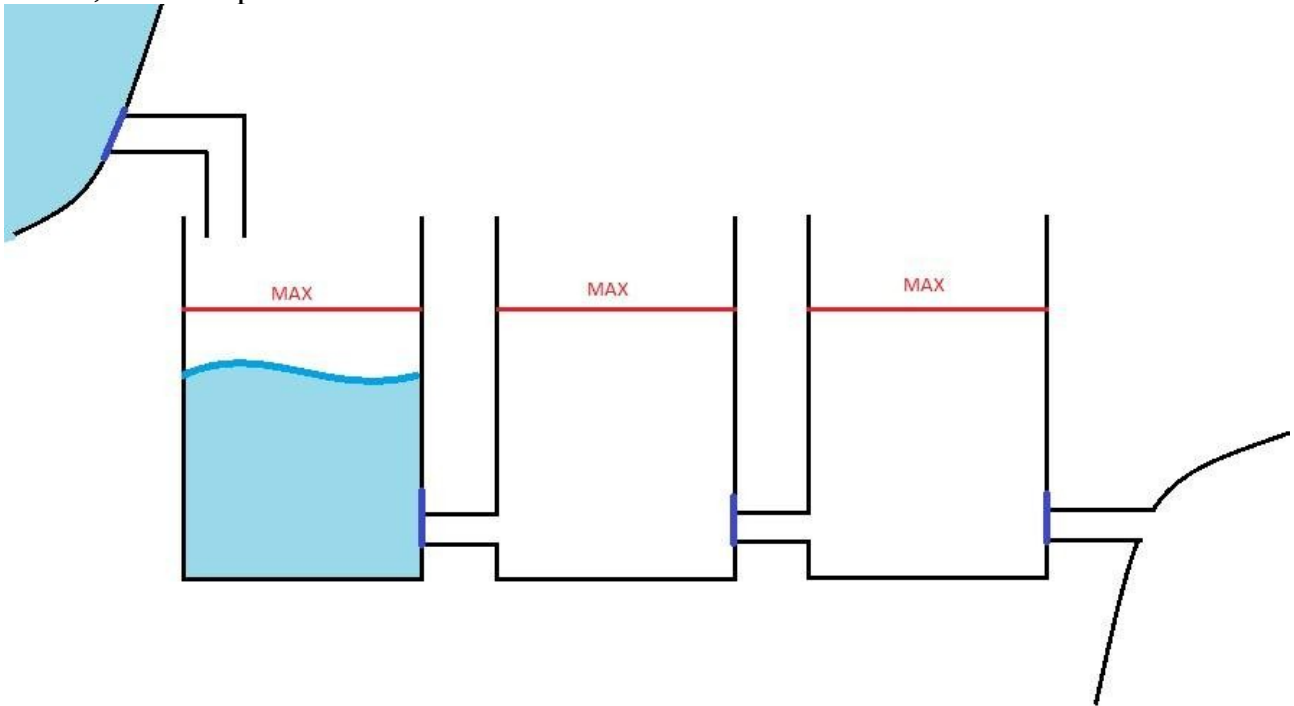


Symulacja systemu regulacji wody w zbiornikach

1.Opis problemu

Projekt systemu regulacji wody w zbiornikach obrazuje jak wygląda proces wlewu wody do zbiornika, a następnie częściowe pozbywanie się go do kolejnych zbiorników. W momencie przepełnienia jednego ze zbiorników otwiera się zawór pośredniczący między danym zbiornikiem a zbiornikiem następnym w układzie w celu przelania nadmiaru wody. W momencie przepełnienia wszystkich zbiorników, część wody uchodzi do odpływu.

Cały system podzielony jest na 3 zbiorniki oraz wlew, są one agentami porozumiewającymi się między sobą. W projekcie jest mowa również o „Odpływie awaryjnym” jednak jest to rzecz umowna. Pokróćce omówimy kod programu, co jak działa, za co odpowiada.



2.Wykorzystane technologie

- JADE(Java Agent DEvelopment Framework)

3.Sposób rozwiązania problemu

Cały program podzielony jest na 4 agentów, kolejno „Wlew”, „Zbiornik1”, „Zbiornik2”, „Zbiornik3”.

Umowna ilość dostarczanej wody z „Wlewu” do zbiornika pierwszego wynosi 100l. Agent ten dostarcza odpowiedniej ilości wody do agenta „Zbiornik1”. Odbywa się to w nieskończonej pętli aż do momentu otrzymania informacji od „Zbiornik1” innej niż 'OK'. W tym momencie agent „Wlew” zatrzymuje swój proces, a do działania

przystępuje pętla zawarta w agencie w którym woda osiągnęła wartość większą niż maksymalna.

Agent „Zbiornik1” odpowiada zaś za proces działania pierwszego zbiornika. W nieskończonej pętli dostarczana jest do zbiornika woda. Proces trwa aż do momentu przekroczenia maksymalnej dopuszczalnej ilości wody w zbiorniku.

```
super.setup();
int zawartosc=0;
int max=1000;
ACLMessage wlew = new ACLMessage(ACLMessage.INFORM);
wlew.addReceiver(new AID("Wlew", AID.ISLOCALNAME));

System.out.println("Zbiornik1 <---> Zbiornik2");

while(true) {
    ACLMessage woda = receive();
    if (woda != null) {
        int dolana=Integer.parseInt(woda.getContent());
        if (zawartosc < max) {
            zawartosc += dolana;
            System.out.println("Zbiornik1 [" + zawartosc + "l]");
            wlew.setContent("OK");
            send(wlew);
        }
    }
}
```

Gdy ilość ta osiągnie maksimum, do agenta „Wlew” dostarczana jest wiadomość 'CLOSE', który wraz z otrzymaniem wiadomości zaprzestaje swoich czynności.

Dalsza część kodu odpowiada za otwarcie zaworu łączącego dwa pierwsze zbiorniki. Tworzymy ACLMessage pwoda (przelana woda), który odpowiada za odprowadzenie połowy wody, plus tej ilości która znalazła się ponad maksimum w zbiorniku pierwszym do zbiornika drugiego. Po przelaniu połowy ilości wody zawór między zbiornikami ulegnie zamknięciu, a między zbiornikiem pierwszym a wlewem otworzy się ponownie.

```

} else {
    zawartosc += dolana;
    System.out.println("Zbiornik1 MAX!");
    //zamknij wlew
    wlew.setContent("CLOSE");
    send(wlew);

    //przelej połowę
    ACLMessage pwoda = new ACLMessage(ACLMessage.INFORM);
    pwoda.addReceiver(new AID("Zbiornik2", AID.ISLOCALNAME));
    while(true){
        pwoda.setContent(String.valueOf(dolana));
        send(pwoda);
        System.out.println("Zbiornik1 >" + dolana + "<" + "Zbiornik2");
        while(true){
            ACLMessage status = receive();
            if (status != null) {
                if (!Objects.equals(status.getContent(), "OK"))
                    System.out.println("Zbiornik1 <-X-> Zbiornik2");
                else
                    break;
            }
        }
        zawartosc -= dolana;
        System.out.println("Zbiornik1 [" + zawartosc + "]");
    }
}

```

Agent „Zbiornik2”, jak i „Zbiornik3” działają podobnie co do agenta „Zbiornik1” z tą różnicą, że ostatni z agentów nadmiar wody wylewa do odpływu awaryjnego, poza tym, woda ta jest wylewana całkowicie, a nie tylko połowicznie z tego zbiornika.

```

} else {
    zawartosc += dolana;
    System.out.println("Zbiornik1 MAX!");
    //zamknij wlew
    wlew.setContent("CLOSE");
    send(wlew);

    //przelej całość
    while(true){
        zawartosc -= dolana;
        System.out.println("Zbiornik3 >" + dolana + "<" + "Odpływ awaryjny");
        System.out.println("Zbiornik3 [" + zawartosc + "]");
        try {
            sleep(1000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        if(zawartosc == 0)
            break;
    }
}

```

4. Wyniki

Dzięki temu prostemu przykładowi symulacji regulacji wody w układzie zbiorników możemy zaobserwować, jak wielkie możliwości daje środowisko JADE. Agenty zawarte w programie wymieniają się danymi bez najmniejszych problemów, stany zbiorników się zgadzają a regulacja przebiega pomyślnie. Oczywiście zjawisko

można by jeszcze pokazać graficznie (np. dzięki zaimplementowaniu specjalnej biblioteki), lecz samo narzędzie sniffera w interfejsie JADE ukazuje przebieg regulacji równie dobrze.