

Санкт-Петербургский Национальный Исследовательский Университет  
Информационных Технологий, Механики и Оптики

Факультет ПИиКТ

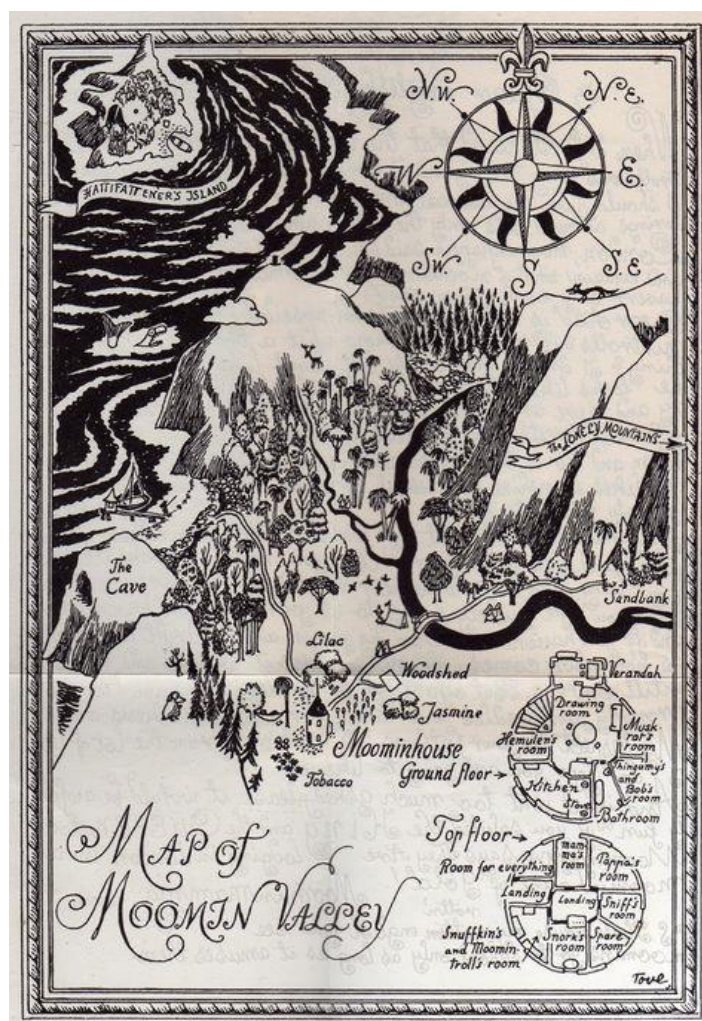
Лабораторная работа №3  
по дисциплине  
«Программирование»

Вариант 971

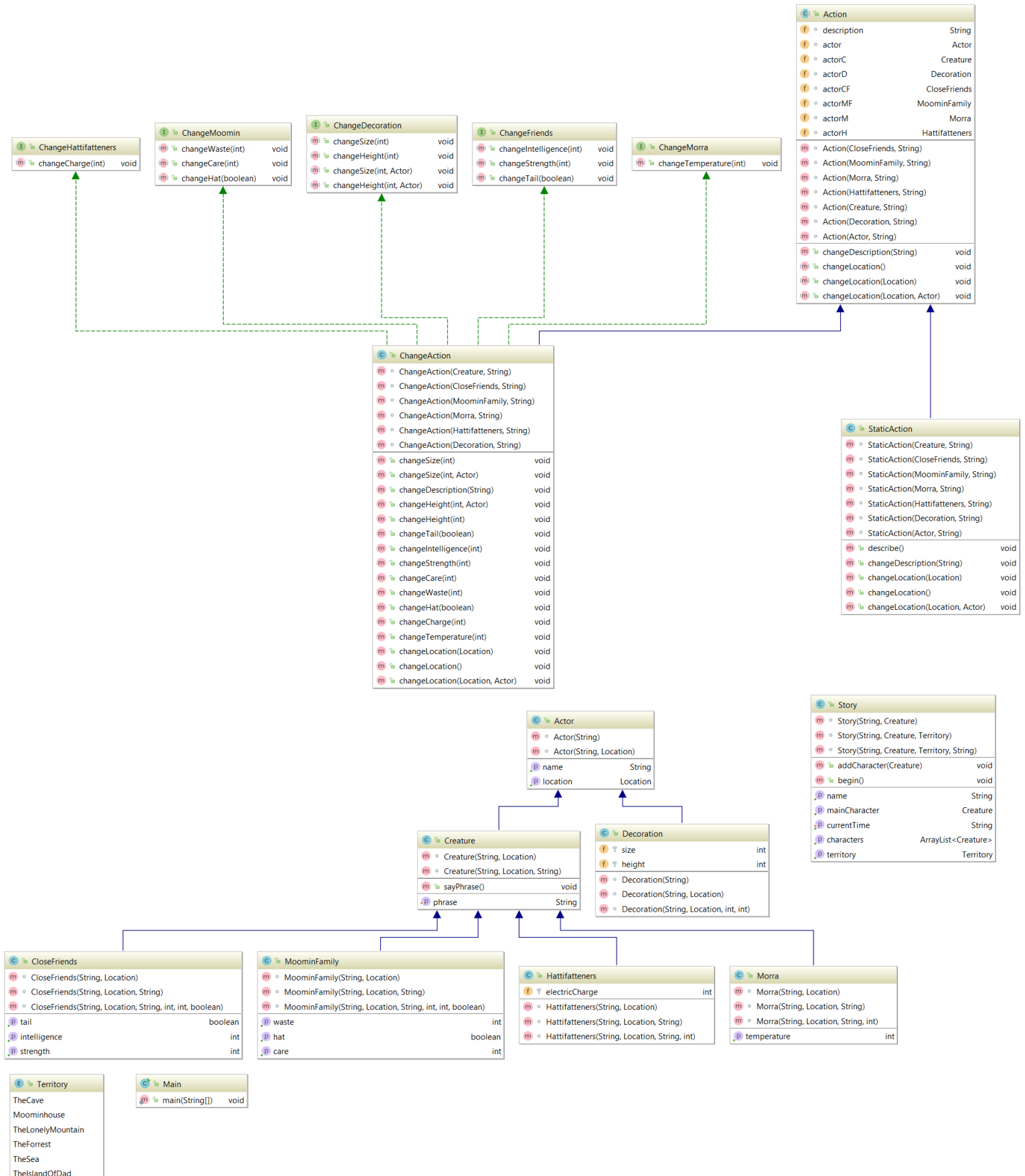
Выполнил: Киселёв Сергей Владимирович  
Проверил: Письмак Алексей Евгеньевич

## Задание

А он торчал на дворе в мокром платье, прилипавшем к ногам, и караулил Панталошку. Время от времени он съедал яблочко или считал тычинки в каком-нибудь цветке джунглей, но больше вздыхал. Дождь перестал, близились сумерки. И едва только зашло солнце, что-то случилось с зеленым холмом, заключившим в себя Муми-дом. Он стал увядать так же быстро, как вырос. Плоды сморщились и попадали на землю. Цветы поникли, а их листья свернулись трубочками. Дом наполнился шорохом и потрескиванием. Хемуль глядел, глядел, потом подошел и легонько потянул к себе ветку. Она была сухая, как трут, и сразу же отломилась: Тут Хемуля осенило. Он собрал огромную кучу хвороста, сходил в дровяной сарай за спичками -- и на садовой дорожке запыхал костер. Веселый и довольный, Хемуль подсел к огню и просушил свое платье. А немного погодя его еще раз осенило, и он с нехемульской силой затащил в огонь хвост Панталошки: больше всего на свете Хемуль любил жареную рыбу.



# UML



## Story.java

```
package com.kamikadze328.vk;

import java.util.ArrayList;

public class Story {
    private String name;
    private ArrayList<Creature> characters= new ArrayList<Creature>();
    private Creature mainCharacter;
    private Territory territory;
    private String currentTime;
    Story(String n, Creature M){
        name = n;
        mainCharacter = M;
        characters.add(M);
        territory = Territory.Moominhouse;
        currentTime = "Day";
    }
    Story(String n, Creature M, Territory t){
        name = n;
        mainCharacter = M;
        characters.add(M);
        territory = t;
        currentTime = "Day";
    }
    Story(String n, Creature M, Territory t, String ct){
        name = n;
        mainCharacter = M;
        characters.add(M);
        territory = t;
        currentTime = ct;
    }

    public void setCurrentTime(String currentTime) {
        this.currentTime = currentTime;
        System.out.print(currentTime + "\n");
    }
    public void addCharacter(Creature creature){
        characters.add(creature);
    }

    public ArrayList<Creature> getCharacters() {
        return characters;
    }

    public String getName() {
        return name;
    }

    public Creature getMainCharacter() {
        return mainCharacter;
    }

    public String getCurrentTime() {
        return currentTime;
    }

    public Territory getTerritory() {
        return territory;
    }
    public void begin(){
        System.out.print("В далёкой-далёкой галактике жил-был " +
mainCharacter.name + "\n");
        System.out.print("Было у него много друзей" + "\n");
        for (int i = 0; i < characters.size(); i++){
            if(characters.get(i).name!=mainCharacter.name) {
                System.out.print(characters.get(i).name + "\n");
            }
        }
    }
}
```

```

    }
    System.out.print("И вот однажды " + currentTime + " в " + territory +
"... \n");
}
}

```

### ChangeMorra.java

```

package com.kamikadze328.vk;

public interface ChangeMorra {
    void changeTemperature(int t);
}

```

### ChangeMoomin.java

```

package com.kamikadze328.vk;

public interface ChangeMoomin {
    void changeWaste(int w);
    void changeCare(int c);
    void changeHat(boolean h);
}

```

### ChangeFriends.java

```

package com.kamikadze328.vk;

public interface ChangeFriends {
    void changeIntelligence(int i);
    void changeStrength(int s);
    void changeTail(boolean t);
}

```

### ChangeHattifatteners.java

```

package com.kamikadze328.vk;

public interface ChangeHattifatteners {
    void changeCharge(int c);
}

```

### ChangeDecoration.java

```

package com.kamikadze328.vk;

public interface ChangeDecoration {
    void changeSize(int s);
    void changeHeight(int h);
    void changeSize(int s, Actor a);
    void changeHeight(int h, Actor a);
}

```

### Action.java

```

package com.kamikadze328.vk;
import com.kamikadze328.vk.*;

public abstract class Action {
    String description;
    Actor actor;
    Creature actorC;
    Decoration actorD;
    CloseFriends actorCF;
    MoominFamily actorMF;
    Morra actorM;
    Hattifatteners actorH;

    Action(CloseFriends a, String d){
        actorCF = a;
        description = d;
    }
}

```

```

    Action(MoominFamily a, String d){
        actorMF = a;
        description = d;
    }
    Action(Morra a, String d){
        actorM = a;
        description = d;
    }
    Action(Hattifatteners a, String d){
        actorH = a;
        description = d;
    }
    Action(Creature a, String d){
        actorC = a;
        description = d;
    }
    Action(Decoration a, String d){
        actorD = a;
        description = d;
    }
    Action(Actor a, String d){
        actor = a;
        description =d;
    }
    public void changeDescription(String s){
        description = s;
    }
    public abstract void changeLocation();
    public abstract void changeLocation(Location l);
    public abstract void changeLocation(Location l, Actor a);
}

```

### StaticAction.java

```

package com.kamikadze328.vk;
import com.kamikadze328.vk.*;

public class StaticAction extends Action{
    StaticAction(Creature a, String d){
        super(a, d);
    }
    StaticAction(CloseFriends a, String d){
        super(a, d);
    }
    StaticAction(MoominFamily a, String d){
        super(a, d);
    }
    StaticAction(Morra a, String d){
        super(a, d);
    }
    StaticAction(Hattifatteners a, String d){
        super(a, d);
    }
    StaticAction(Decoration a, String d){
        super(a, d);
    }
    StaticAction(Actor a, String d){
        super(a,d);
    }
    public void describe(){
        if(actorD!=null){
            System.out.println(actorD.name + " " + description);
        }
        if(actorCF!=null){
            System.out.println(actorCF.name + " " + description);
        }
        if(actorC!=null){
            System.out.println(actorC.name + " " + description);
        }
    }
}

```

```

        if(actorMF!=null) {
            System.out.println(actorMF.name + " " + description);
        }
        if(actorM!=null) {
            System.out.println(actorM.name + " " + description);
        }
    }

    @Override
    public void changeDescription(String s) {
        super.changeDescription(s);
    }

    @Override
    public void changeLocation(Location l) {
        if(actorD!=null&&actorD.location==1) {
            System.out.println(actorD.name + " " + description + " B " + 1);
        }
        if(actorCF!=null&&actorCF.location==1) {
            System.out.println(actorCF.name + " " + description + " B " + 1);
        }
        if(actorC!=null&&actorC.location==1) {
            System.out.println(actorC.name + " " + description + " B " + 1);
        }
        if(actorMF!=null&&actorMF.location==1) {
            System.out.println(actorMF.name + " " + description + " B " + 1);
        }
        if(actorM!=null&&actorM.location==1) {
            System.out.println(actorM.name + " " + description + " B " + 1);
        }
    }

    @Override
    public void changeLocation() {
        if(actorD!=null) {
            System.out.print(actorD.name + " " + description + " B " +
actorD.location + "\n");
        }
        if(actorCF!=null) {
            System.out.print(actorCF.name + " " + description + " B " +
actorCF.location + "\n");
        }
        if(actorC!=null) {
            System.out.print(actorC.name + " " + description + " B " +
actorC.location + "\n");
        }
        if(actorMF!=null) {
            System.out.print(actorMF.name + " " + description + " B " +
actorMF.location + "\n");
        }
        if(actorM!=null) {
            System.out.print(actorM.name + " " + description + " B " +
actorM.location + "\n");
        }
    }

    @Override
    public void changeLocation(Location l, Actor a) {
        if(actorD!=null&&actorD.location==1) {
            System.out.println(a.name + " " + description + " " + actorD.name + " B
" + 1);
        }
        if(actorCF!=null&&actorCF.location==1) {
            System.out.println(a.name + " " + description + " " + actorCF.name + " B
" + 1);
        }
        if(actorC!=null&&actorC.location==1) {
            System.out.println(a.name + " " + description + " " + actorC.name + " B
" + 1);
        }
    }

```

```

    }
    if(actorMF!=null&&actorMF.location==1){
        System.out.println(a.name + " " + description + " " + actorMF.name + " в
" + 1);
    }
    if(actorM!=null&&actorM.location==1){
        System.out.println(a.name + " " + description + " " + actorM.name + " в
" + 1);
    }
}
}

```

## ChangeAction.java

```

package com.kamikadze328.vk;
import com.kamikadze328.vk.*;

```

```

public class ChangeAction extends Action implements ChangeFriends,
ChangeHattifatteners, ChangeMoomin, ChangeMorra, ChangeDecoration{
    ChangeAction(Creature a, String d){
        super(a, d);
    }
    ChangeAction(CloseFriends a, String d){
        super(a, d);
    }
    ChangeAction(MoominFamily a, String d){
        super(a, d);
    }
    ChangeAction(Morra a, String d){
        super(a, d);
    }
    ChangeAction(Hattifatteners a, String d){
        super(a, d);
    }
    ChangeAction(Decoration a, String d){
        super(a, d);
    }

    @Override
    public void changeSize(int s) {
        System.out.printf(actorD.name + " " + description + " до размера %d \n",
s);
        actorD.size = s;
    }

    @Override
    public void changeSize(int s, Actor a) {
        System.out.printf(a.name + " " + description + " до размера "+ s + "
предмет "+actorD.name+"\n");
    }

    @Override
    public void changeDescription(String s) {
        super.changeDescription(s);
    }

    @Override
    public void changeHeight(int h, Actor a) {
        if(h>actorD.height){
            System.out.printf(a.name + " поднимает " + actorD.name + " на %d\n",
h-actorD.height);
        }
        if(h<actorD.height&&h==0){
            System.out.print(a.name + " кладёт " + actorD.name + " на землю\n");
        }
        else{
            System.out.printf(a.name + " опускает " + actorD.name + " на %d\n",
-h+actorD.height);
        }
    }
}

```



```

    }
}

@Override
public void changeHeight(int h) {
    if(h>actorD.height){
        System.out.printf(actorD.name + " поднялся на %d\n", h-
actorD.height);
    }
    if(h<actorD.height&&h==0){
        System.out.print(actorD.name + " упал на землю\n");
    }
    else{
        System.out.printf(actorD.name + " опустился на %d\n", -
h+actorD.height);
    }
    actorD.height = h;
}

@Override
public void changeTail(boolean t) {
    actorCF.tail = t;
}

@Override
public void changeIntelligence(int i) {
    actorCF.intelligence = i;
}

@Override
public void changeStrength(int s){
    actorCF.strength = s;
}

@Override
public void changeCare(int c) {
    actorMF.care = c;
}

@Override
public void changeWaste(int w) {
    actorMF.waste = w;
}

@Override
public void changeHat(boolean h) {
    actorMF.hat = h;
}

@Override
public void changeCharge(int e) { actorH.electricCharge = e; }

@Override
public void changeTemperature(int t) { actorM.temperature = t; }

@Override
public void changeLocation(Location l) {
    if (actorD != null && actorD.location != 1) {
        actorD.location = 1;
        System.out.print(actorD.name + " " + description + " в " + l +
"\n");
    }
    if (actorCF != null && actorCF.location != 1) {
        actorCF.location = 1;
        System.out.print(actorCF.name + " " + description + " в " + l +
"\n");
    }
    if (actorC != null && actorC.location != 1) {
        actorC.location = 1;

```

```

        System.out.print(actorC.name + " " + description + " B " + l +
"\n");
    }
    if (actorMF != null && actorMF.location != 1) {
        actorMF.location = 1;
        System.out.print(actorMF.name + " " + description + " B " + l +
"\n");
    }
    if (actorM != null && actorM.location != 1) {
        actorM.location = 1;
        System.out.print(actorM.name + " " + description + " B " + l +
"\n");
    }
}
}

```

## Actor.java

```

package com.kamikadze328.vk;

import com.kamikadze328.vk.*;
import java.util.ArrayList;

public class Actor {
    protected String name;
    protected Location location;
    Actor(String n){
        name = n;
    }
    Actor(String n, Location l){
        name = n;
        location = l;
    }

    public String getName(){
        return name;
    }
    public Location getLocation(){
        return location;
    }
}

```

## Decoration.java

```

package com.kamikadze328.vk;

public class Decoration extends Actor {
    protected int size;
    protected int height;
    Decoration(String n){
        super(n);
    }
    Decoration(String n, Location l){
        super(n,l);
    }
    Decoration(String n, Location l, int s, int h){
        super(n,l);
        size = s;
        height = h;
    }
}

```

## Creature.java

```

package com.kamikadze328.vk;

import com.kamikadze328.vk.Actor;

public class Creature extends Actor {
    private String phrase;
}

```

```

Creature(String n, Location l){
    super(n,l);
}
Creature(String n, Location l, String p){
    super(n,l);
    phrase = p;
}
public void setPhrase(String aPhrase){
    phrase = aPhrase;
}
public void sayPhrase(){
    System.out.print(name + ": " + phrase + "\n");
}
}

```

## CloseFriends.java

**package** com.kamikadze328.vk;

```

public class CloseFriends extends Creature {
    protected int intelligence;
    protected int strength;
    protected boolean tail;
    CloseFriends(String n, Location l){
        super(n,l);
    }
    CloseFriends(String n, Location l, String p) {
        super(n, l, p);
    }
    CloseFriends(String n, Location l, String p, int i, int s, boolean t){
        super(n,l,p);
        intelligence = i;
        strength = s;
        tail = t;
    }

    public int getIntelligence() {
        return intelligence;
    }

    public int getStrength() {
        return strength;
    }
    public boolean getTail(){
        return tail;
    }
}

```

## Hatifatteners.java

**package** com.kamikadze328.vk;

```

public class Hattifatteners extends Creature {
    protected int electricCharge;
    Hattifatteners(String n, Location l){
        super(n,l);
    }
    Hattifatteners(String n, Location l, String p) {
        super(n, l, p);
    }
    Hattifatteners(String n, Location l, String p, int e){
        super(n,l,p);
        electricCharge = e;
    }
}

```

## Morra.java

```

public class Morra extends Creature {
    protected int temperature;
    Morra(String n, Location l){
        super(n,l);
    }
    Morra(String n, Location l, String p) {
        super(n, l, p);
    }
    Morra(String n, Location l, String p, int t){
        super(n,l,p);
        temperature = t;
    }

    public int getTemperature() {
        return temperature;
    }
}

```

### MoominFamily.java

```
package com.kamikadze328.vk;
```

```

public class MoominFamily extends Creature{
    protected int waste;
    protected int care;
    protected boolean hat;
    MoominFamily(String n, Location l){
        super(n,l);
    }
    MoominFamily(String n, Location l, String p) {
        super(n, l, p);
    }
    MoominFamily(String n, Location l, String p, int w, int c, boolean h){
        super(n,l,p);
        waste = w;
        care = c;
        hat = h;
    }

    public int getCare() {
        return care;
    }

    public int getWaste() {
        return waste;
    }
    public boolean getHat(){
        return hat;
    }
}

```

### Location.java

```
package com.kamikadze328.vk;
```

```

public enum Location {
    TopFloor,
    GroudFloor,
    Yard,
    Hill,
    Hangar,
    FootPath,
    Seamark,
    NaN
}

```

### Territory.java

```
package com.kamikadze328.vk;
```

```
public enum Territory {  
    TheCave,  
    Moominhouse,  
    TheLonelyMountain,  
    TheForrest,  
    TheSea,  
    TheIslandOfDad  
}
```

## Main.java

```
package com.kamikadze328.vk;
```

```
import com.kamikadze328.vk.*;
```

```
public class Main {
```

```
    public static void main(String[] args) {  
        CloseFriends Khemul = new CloseFriends("Хемуль", Location.Hill,  
        "*вздых*", 499, 50, false);  
        StaticAction karaul = new StaticAction(Khemul, "караулит Панталошку");  
        StaticAction yabloko = new StaticAction(Khemul, "ест яблоко");  
        StaticAction schitat = new StaticAction(Khemul, "считает тычинки в  
цветке джунглей");  
        StaticAction smotret = new StaticAction(Khemul, "смотрит");  
        StaticAction osenit = new StaticAction(Khemul, "выполняет мыслительные  
процессы");  
        StaticAction dozhd = new StaticAction(new Decoration("Дождь"),  
        "перестал");  
        ChangeAction sun = new ChangeAction(new Decoration("Солнце"), "зашло");  
        Story moomin = new Story("kek", Khemul, Territory.Moominhouse,  
        "вечером");  
        ChangeAction holm = new ChangeAction(new Decoration("холм",  
Location.Hill, 10, 0), "увядает");  
        ChangeAction plod = new ChangeAction(new Decoration("плод",  
Location.Hill, 3, 7), "сморщивается");  
        ChangeAction tsvetok = new ChangeAction(new Decoration("цветок",  
Location.Hill, 2, 2), "пониж");  
        StaticAction list = new StaticAction(new Decoration("лист"),  
        "сворачивается трубочкой");  
        Creature dom = new Creature("дом", Location.Hill, "*шорох и  
потрескивание*");  
        ChangeAction hodit = new ChangeAction(Khemul, "подходит к кусту");  
        Decoration vetka = new Decoration("ветка", Location.Hill, 1, 7);  
        ChangeAction vetkaC = new ChangeAction(vetka, "");  
        StaticAction vetkaS = new StaticAction(vetka, "ломается");  
        Decoration vetki = new Decoration("ветки");  
        ChangeAction sobrat = new ChangeAction(vetki, "собирает");  
        ChangeAction skhodit = new ChangeAction(Khemul, "идёт");  
        StaticAction brat = new StaticAction(Khemul, "берёт спички");  
        StaticAction koster = new StaticAction(new Decoration("костёр",  
Location.FootPath, 6, 0), "пылает");  
        Decoration Khvost = new Decoration("хвост Панталошки",  
Location.FootPath, 5, 0);  
        ChangeAction sila = new ChangeAction(Khemul, "увеличивает");  
        ChangeAction tashit = new ChangeAction(Khvost, "тащит");  
  
        moomin.begin();  
        Khemul.sayPhrase();  
        if (Math.random() > 0.5) {  
            yabloko.describe();  
        }  
        else {  
            schitat.describe();  
        }  
        moomin.setCurrentTime("близятся сумерки");  
        dozhd.describe();  
        sun.changeLocation();  
        holm.changeSize(5);  
    }
```

```

        plod.changeSize(2);
        plod.changeHeight(0);
        tsvetok.changeSize(1);
        list.describe();
        dom.sayPhrase();
        smotret.describe();
        hodit.changeLocation();
        vetkaC.changeHeight(5, Khemul);
        vetkaS.describe();
        osenit.describe();
        sobrat.changeLocation(Location.Yard, Khemul);
        skhodit.changeLocation(Location.Hangar, Khemul);
        brat.describe();
        skhodit.changeLocation(Location.FootPath, Khemul);
        koster.describe();
        sila.changeStrength(10000);
        tashit.changeLocation(Location.Yard, Khemul);
    }
}

```

## Вывод программы

В далёкой-далёкой галактике жил-был Хемуль  
 Было у него много друзей  
 И вот однажды вечером в Moominhouse...

Хемуль: \*вздых\*

Хемуль ест яблоко

близятся сумерки

Дождь перестал

Солнце зашло

холм увядает до размера 5

плод сморщивается до размера 2

плод упал на землю

цветок поник до размера 1

лист сворачивается трубочкой

дом: \*шорох и потрескивание\*

Хемуль смотрит

Хемуль подходит к кусту

Хемуль опускает ветка на 2

ветка ломается

Хемуль выполняет мыслительные процессы

Хемуль собирает ветки в Yard

Хемуль идёт Хемуль в Hangar

Хемуль берёт спички

Хемуль идёт Хемуль в FootPath

костёр пылает

Хемуль тащит хвост Панталочки в Yard

Вывод: В далёкой-далёкой галактике жил-был мальчик, который сдавал все лабы  
 вовремя... Однако история умалчивает, что с ним случилось.