Институт информационных технологий и управления

в технических системах

Кафедра информационных технологий и компьютерных систем

ОТЧЕТ

по лабораторной работе № 6

«**ИССЛЕДОВАНИЕ РАЗВЕТВЛЯЮЩИХСЯ АЛГОРИТМОВ**»

по дисциплине «Программирование. Базовые процедуры обработки информации»

Выполнил студент группы ИВТ/б-11д

Орлов И.В.

Проверил доцент Петров И.И.

Севастополь

2020

**1. ЦЕЛЬ РАБОТЫ**

Целью данной работы является исследование разветвляющихся алгоритмов и их программирование с помощью условных операторов языка Java, приобретение начальных навыков тестирования программ.

2. **ПОСТАНОВКА ЗАДАЧИ**

1) Изучить возможности условных операторов Java и ознакомиться с примерами программ, приведенными в п. 2.7 методических указаний.

2) Разработать и отладить программы, демонстрирующие применение условных операторов if и switch.

3) Протестировать программы на контрольных примерах.

Вариант задания V вычислен по формуле

V = (N%14 != 0) ? N%14 : 14; ,

где N – номер студента в списке группы:

25%14=11;

В качестве первого индивидуального задания на лабораторную работу предлагается разработать программу, реализующую вычисление y=g(x)+z(x) при заданном значении x. Формулы для вычисления g и z приведены в таблице 2.1.

В качестве второго индивидуального задания на лабораторную работу предлагается разработать программу, реализующую выбор одного из нескольких вариантов действий в зависимости от значения переменной choice. Вариант второго задания приведен в таблице 2.2.

Таблица 2.1 – Вариант первого индивидуального задания

|  |  |  |
| --- | --- | --- |
| Номер  варианта | g | z |
| 11 |  |  |

Таблица 2.2 – Вариант второго индивидуального задания

|  |  |
| --- | --- |
| Номер варианта | Содержание задания |
| 11 | Преобразовать температуру воздуха (в градусах по Цельсию) в характеристику погоды теплого времени года по правилу: 3 – 7 : "холодная"; 8 – 12: "прохладная"; 13 – 20: "теплая"; 21 – 28: "очень теплая"; 29 – 35: "жаркая"; 36 – 42: "очень жаркая"; иначе : если температура меньше трех градусов, то сообщение «Вот и зима пришла ;-)», иначе – «крайне жаркая». |

**3. ВЫПОЛНЕНИЕ ПЕРВОГО ИНДИВИДУАЛЬНОГО ЗАДАНИЯ**

**3.1. Анализ задачи**

Необходимо разработать программу, реализующую вычисление y=g(x)+z(x) при заданном значении x. Таким образом, входным данным программы является значение переменной x, выходными данными программы являются значения переменных g, z и y. Заданы математические формулы для вычисления функций g и z, причем значения функций вычисляются по различным формулам для различных интервалов области определения функций. Для вычисления функции g можно использовать троичную условную операцию Java, т.к. существует два варианта вычисления функции (даны формулы для областей x<=0 и x>0) .

Для вычисления функции z целесообразно использовать форму if-else-if условного оператора if, т.к. существует три взаимоисключающих варианта вычисления функции (даны формулы для областей определения x<0, x>=1, 0<=x<1).

Выходные данные (значения переменных g, z и y при заданном значении переменной x) будут выводиться в окно терминала с помощью метода форматной печати System.out.printf().

Поскольку в условии даны непрерывные области определения функции, а также в виду того, что необходимые для вычислений функции модуля Math работают с double-значениями, представляется целесообразным для переменных x, g, z и y выбрать тип double. Кроме того, не рекомендуется смешивать значения разных типов в арифметических выражениях, чтобы избежать ошибок в вычислениях.

**3.2. Тестовые примеры и результаты его обработки вручную**

Для проверки всех ветвей разветвляющегося вычислительного процесса потребуется три теста, т.е. нужно вычислить значение функций g, z и y в трех точках, каждая из которых принадлежит соответствующей области определения (одной из трех, указанных выше).

Далее нужно выбрать три значения x из разных областей определения функции z, привести результаты расчета на калькуляторе значений функций g, z, y для каждого из значений х.

**3.3. Схема алгоритма вычислительного процесса**

Здесь нужно привести алгоритм вычислений (будет иметь разветвляющуюся структуру), правильно оформив его в виде рисунка (рисунок 3.1) с подрисуночной подписью и ссылкой на этот рисунок в тексте отчета (требования к оформлению изложены в п. 2.8.1 части первой методических указаний). Пример схемы алгоритма приведен в п. 2.7.1 методических указаний на рисунке 2.2. На схеме алгоритма нужно показать пути реализации каждого из трех тестов.

Рисунок 3.1 – Схема алгоритма вычислительного процесса

для задания 1

**3.4. Текст программы**

Согласно пункту 2.8.1. методических указаний к лабораторным работам, текст программы должен быть отформатирован следующим образом:

1. шрифт: arial, 12 пт, полужирный;
2. абзац:

* выравнивание по левому краю;
* уровень – основной текст;
* отступ слева – 0;
* отступ справа – 0;
* интервал перед – 0;
* интервал после – 0;
* первая строка – нет (отступа или выступа),
* междустрочный интервал – множитель, значение – 1,2;

1. комментарии (обязательно должны присутствовать в тексте программы) выделить курсивом и синим цветом.

**3.5. Сведения об отладке программы и проверке ее работоспособности**

Результат выполнения программы на первом тестовом примере изображен на рисунке 3.2.

Скриншот окна терминала

(размер рисунка должен быть достаточным для того, чтобы информацию рисунка было легко воспринимать).

Рисунок 3.2 – Результат выполнения программы

на первом тестовом примере

Результат работы программы на втором тестовом примере изображен на рисунке 3.3.

Скриншот окна терминала

(размер рисунка должен быть достаточным для того, чтобы информацию рисунка было легко воспринимать).

Рисунок 3.3 – Результат выполнения программы

на втором тестовом примере

Результат работы программы на третьем тестовом примере изображен на рисунке 3.4.

Скриншот окна терминала

(размер рисунка должен быть достаточным для того, чтобы информацию рисунка было легко воспринимать).

Рисунок 3.4 – Результат выполнения программы

на третьем тестовом примере

На тестовых наборах входных данных программа выдала ожидаемый результат, что позволяет сделать вывод о ее работоспособности.

**4. ВЫПОЛНЕНИЕ ВТОРОГО ИНДИВИДУАЛЬНОГО ЗАДАНИЯ**

**4.1. Анализ задачи**

Вариант задания (таблица 2.2) предписывает на основе анализа целочисленного значения (температуры воздуха) выбрать выполнение определенного действия (присвоение характеристике погоды определенного строкового значения), причем одно и то же действие (одна и та же характеристика погоды) может соответствовать нескольким целочисленным значениям (температуры). Поскольку альтернативы целочисленные, число их конечно и больше двух для решения задачи представляется целесообразным использовать оператор switch c множественными ветвями case. На входе программы – конкретное значение температуры воздуха. Для его хранения в программе будет определена переменная типа int. На выходе программы – характеристика погоды. Для ее хранения потребуется определить переменную класса String (символьная строка). После выполнения оператора switch характеристика погоды должна быть выведена в окно терминала. Для этого можно использовать метод форматированного вывода System.out.printf().

**4.2. Тестовые примеры и результаты их обработки вручную**

Здесь нужно привести по одному значению из каждого интервала значений температуры, которому соответствует определенная характеристика погоды, и указать эту характеристику для соответствующего значения. Тестовые примеры можно оформить в виде таблицы (Таблица 4.1 – Тестовые примеры) со ссылкой на нее в тексте данного пункта (требования к оформлению таблиц изложены в п. 2.8.1 методических указаний).

**4.3. Схема алгоритма вычислительного процесса**

Здесь нужно привести алгоритм вычислений (будет иметь разветвляющуюся структуру), правильно оформив его в виде рисунка (рисунок 4.1) с подрисуночной подписью и ссылкой на этот рисунок в тексте отчета (требования к оформлению изложены в п. 2.8.1 части первой методических указаний). Пример схемы алгоритма приведен в п. 2.7.2 методических указаний на рисунке 2.4.

Рисунок 4.1 – Схема алгоритма вычислительного процесса

для задания 2

**4.4. Текст программы**

Согласно пункту 2.8.1. методических указаний к лабораторным работам, текст программы должен быть отформатирован следующим образом:

1. шрифт: arial, 12 пт, полужирный;
2. абзац:

* выравнивание по левому краю;
* уровень – основной текст;
* отступ слева – 0;
* отступ справа – 0;
* интервал перед – 0;
* интервал после – 0;
* первая строка – нет (отступа или выступа),
* междустрочный интервал – множитель, значение – 1,2;

1. комментарии (обязательно должны присутствовать в тексте программы) выделить курсивом и синим цветом.

**4.5. Сведения об отладке программы и проверке ее работоспособности**

Результат работы программы на первом тестовом примере (п. 4.2), изображен на рисунке 4.2.

Скриншот окна терминала

(размер рисунка должен быть достаточным для того, чтобы информацию рисунка было легко воспринимать).

Рисунок 4.2 – Результат выполнения программы

на первом тестовом примере

Результат работы программы на втором тестовом примере (п.4.2), изображен на рисунке 4.3.

Скриншот окна терминала

(размер рисунка должен быть достаточным для того, чтобы информацию рисунка было легко воспринимать).

Рисунок 4.3 – Результат выполнения программы

на втором тестовом примере

Далее аналогично для остальных тестовых примеров.

На тестовых наборах входных данных программа выдала ожидаемый результат, что позволяет сделать вывод о ее работоспособности.

**ВЫВОД**

В ходе выполнения лабораторной работы были изучены условные операторы языка Java: if и switch, а также разработаны две программы с разветвляющейся структурой, использующие эти операторы.