Министерство науки и высшего образования РФ

ФГАОУ ВО ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИУ)

Институт естественных и точных наук

Факультет математики, механики и компьютерных технологий

Кафедра прикладной математики и программирования

«Планетарная система. Универсальная демонстрационная астрономическая модель»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К ПРОЕКТУ   
по дисциплине «Учебная практика»

ЮУрГУ–01.03.02.2023.11.ПЗ

|  |  |  |
| --- | --- | --- |
|  |  | *Руководитель,*  *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Демидов А.К.*  *«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023г.* |
| *Авторы работы:*  *Студент группы: ЕТ – 112*  *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Бакалова Я.О.*  *Студент группы: ЕТ – 112*  *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Бурматова Д.М.*  *Студент группы: ЕТ – 112*  *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Щипков Т.В.* |  | *Студент группы: ЕТ – 112*  *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Шаталов В.Д.*  *Студент группы: ЕТ – 112*  *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Саломатина П.Д.*  *Студент группы: ЕТ – 112*  *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Хрусталёв Д.А.*  *«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023г.* |
|  |  | *Работа защищена с оценкой*  *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*  *«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023г.* |

Челябинск 2023

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 3](#_Toc140753239)

[1 ПОСТАНОВКА ЗАДАЧИ 4](#_Toc140753240)

[2 РАЗРАБОТКА АЛГОРИТМА 5](#_Toc140753241)

[3 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 6](#_Toc140753242)

[ЗАКЛЮЧЕНИЕ 7](#_Toc140753243)

[БИБЛИОГРАФИЧЕСКИЙ СПИСОК 8](#_Toc140753244)

[ПРИЛОЖЕНИЕ а 9](#_Toc140753245)

# ВВЕДЕНИЕ

**Актуальность темы**. Основными вопросами коллективной разработки программного обеспечения является разделение труда и организация взаимодействия между участниками.

**Цель работы** – разработать программу «Планетарная система. Универсальная демонстрационная астрономическая модель».

**Задачи работы**:

– научиться ставить цели, формулировать задачи индивидуальной и совместной деятельности, решать поставленные задачи в кооперации с коллегами;

– научиться проводить поиск и анализ научной литературы и алгоритмов с последующим исследованием и разработкой математических моделей и программ для предложенных задач;

– ознакомиться с основными видами и задачами будущей профессиональной деятельности.

**Объект работы** – планетарная система. Универсальная демонстрационная астрономическая модель.

**Предмет работы** – применение технологий коллективной разработки программного обеспечения для разработки программы.

**Результаты работы** можно использовать в процессе последующего обучения в соответствии с учебным планом подготовки бакалавров по направлению «Прикладная математика и информатика».

# 1 ПОСТАНОВКА ЗАДАЧИ

Необходимо разработать программу для приложения «Планетарная система. Универсальная демонстрационная астрономическая модель». Для разработки необходимо использовать язык программирования C++ и графическую библиотеку winBGIm.

Программа должна считывать входной файл и вывести анимированную схему планетарной системы. Неподвижная центральная планета помещается в центр экрана.

Программа должна иметь следующие функции:

– масштабирование с помощью колёсика мыши;

– считывание с файла идентификатора планеты, идентификатора базовой планеты, радиуса планеты, радиуса орбиты, угловой скорости планеты;

– отображение комет (рисунок 1);

– отображение звёзд (см. рисунок 1);

– отображение планет, их спутников и орбит (см. рисунок 1).

Примерный интерфейс программы показан на рисунке 1.

Выбор и выполнение действий выполняются с помощью компьютерной мыши.

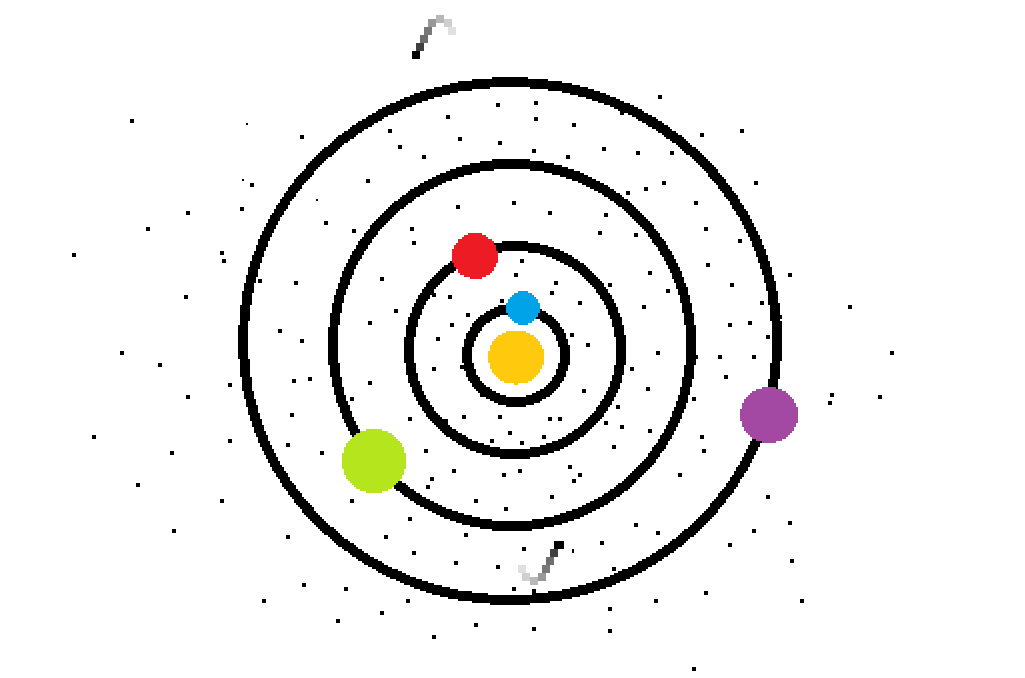


Рисунок 1. – Примерный интерфейс программы «Планетарная система. Универсальная демонстрационная астрономическая модель»

# 2 РАЗРАБОТКА АЛГОРИТМА

Входные данные программы записаны в текстовом файле. Каждая строка описывает одну планету с помощью пяти значений: ID, BASEID, R\_PAD, ORB\_RAD, SPEED.

Основные сущности в программе:

– одномерный массив звёзд stars\_rem, каждая звезда является структурой, содержащей координаты звезды и её цвет;

– вектор планет all\_planets, каждая планета является структурой, содержащей идентификатор планеты, идентификатор планеты, радиус планеты, радиус орбиты планеты, угловая скорость планеты.

– уникальный идентификатор планеты ID – целое число;

– идентификатор базовой планеты BASEID, спутником которой является текущая (описываемая) планета – целое число;

– радиус планеты (в пикселях) R\_PAD – целое число;

– радиус орбиты планеты (в пикселях) ORB\_RAD – целое число;

– угловая скорость планеты SPEED – вещественное число.

Значения BASEID, ORB\_RAD и SPEED для центральной планеты системы (планеты с ID равным 0) игнорируются.

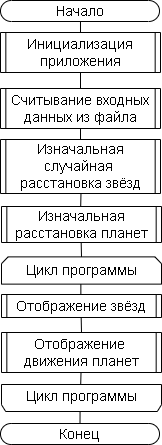
Схема основного алгоритма программы показана на рисунке 2.

Рисунок 2. – Основной алгоритм программы

# 3 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Пользователь вписывает данные о планетах в текстовый файл (рисунок 3).

Рисунок 3. – Входные данные в текстовом файле

При запуске программы открывается окно с планетарной системой, планеты которой отображаются в соответствии с входными данными из файла (рисунок 4).

Рисунок 4. – Планетарная система

Пользователь может масштабировать изображения планет с помощью колесика мыши (рисунок 5).

Рисунок 5. – Масштабирование планетарной системы

При нажатии на крестик в правом верхнем углу окна приложение закрывается.

# ЗАКЛЮЧЕНИЕ

В ходе коллективной работы над проектом были поставлены точные требования к программе, затем были выявлены элементы интерфейса пользователя, разработаны необходимые математические модели, определены и детализированы структуры данных и алгоритмы. После завершения проектирования алгоритмы были реализованы на языке С++. Разработанный код был проверен на контрольных тестах и в код были внесены необходимые исправления. Для программы было разработано руководство пользователя. Таким образом, цель работы была достигнута, задачи – решены.

Результаты работы можно использовать в процессе последующего обучения в форме практических навыков коллективной разработки и организации взаимодействия на всех этапах разработки сложных программных систем.

# БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Лауферман, О. В. Разработка программного продукта: профессиональные стандарты, жизненный цикл, командная работа: учебное пособие / О. В. Лауферман, Н. И. Лыгина. — Новосибирск: НГТУ, 2019. — 75 с. — URL: https://e.lanbook.com/book/152251 (дата обращения: 22.07.2023). .

2. Солдатенко, И. С. Практическое введение в язык программирования Си: учебное пособие / И. С. Солдатенко, И. В. Попов. — Санкт-Петербург: Лань, 2018. — 132 с. — URL: https://e.lanbook.com/book/109619 (дата обращения: 22.07.2023).

3. Графическая библиотека WinBGIm. – URL: https://ipc.susu.ru/20786.html (дата обращения: 22.07.2023).

# ПРИЛОЖЕНИЕ а

А.1 Файл interface.h

#ifndef INTERFACE\_H

#define INTERFACE\_H

// область рисования

#define PAREA\_X 100

#define PAREA\_Y 20

#define PAREA\_W 681

#define PAREA\_H 531

// выбор цветов

#define COLORS\_X 10

#define COLORS\_Y 200

#define COLORS\_D 35

extern int f\_color, // цвет рисования

b\_color, // цвет закраски

tool; // текущий инструмент

extern char pic\_name[100]; // имя файла

void edit\_name(); // ввести имя файла

void set\_xy(int x, int y); // показать текущие координаты в строке состояния

void draw\_colors(); // нарисовать текущие цвета

int in\_parea(int x, int y); // в области рисования?

#endif

А.2 Файл interface.cpp

#include <graphics.h>

#include <string.h>

#include <stdio.h>

#include "interface.h"

int f\_color=BLACK, // цвет рисования

b\_color=WHITE, // цвет закраски

tool=2; // инструмент карандаш

char pic\_name[100]="untitled.bmp";

void draw\_colors() {

int x1=20,y1=120,x2=70, y2=170;

setfillstyle(SOLID\_FILL,f\_color);

bar(x1,y1,x2,y2);

setfillstyle(SOLID\_FILL,b\_color);

...