**КАЗАХСКИЙ НАЦИОНАЛЬНЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ АБАЯ**

**«Программистский калькулятор»**

(проект)

**Выполнил:** Злавдинов С. М.

**Специальность:** 6В01509 –

Информатика и робототехника

**Руководитель:** Киселёва Е. А.

**Алматы-2022**

**Оглавление**

|  |  |
| --- | --- |
| Паспорт проектной работы | 2 |
| Введение | 3 |
| Глава 1. Язык программирования Java. | 4 |
| 1.1 Краткая информация о Системе счисления. | 4 |
|  |  |
| Глава 2. Описание работы кода. | 5 |
| 2.1 Основные свойства, используемые при счислении. | 5 |
| 2.2 Разбор кода по частям. | 5 |
| Заключение  Список литературы | 9  10 |

**Паспорт проектной работы**

|  |  |
| --- | --- |
| Название проекта | Программистский калькулятор |
| Участник проекта | Злавдинов Султанали |
| Цель проекта | Написать калькулятор с элементами системы счисления |
| Задачи проекта | 1. Создать обычный калькулятор 2. Добавить функции счисления |

**Введение**

Рутинное времяпровождение в городе. Гул на перекрестках города, работа, вечные проблемы и заботы... И все это каждый день. Не было бы человека, кому бы это не надоело, и у каждого возникает желание почувствовать себя свободным от всех забот и проблем, хотя бы на несколько минут. Но не всегда получается выделить для отдыха большое количество времени. И тогда, на помощь "уставшим" приходят современные компьютеры. Они дают возможность "уйти" от "серых" будней и погрузиться в игровую реальность, которая дает возможность почувствовать себя свободным и не обремененным заботами, при этом не заставляя тратить время или деньги на дорогу, и всегда оставляя возможность максимально быстро вернуться к работе.

Развитие человечества все больше и больше наталкивало величайшие умы на мысли об автоматизации некоторых процессов. Можно с уверенностью утверждать, что создание данного кода на языке программирования Java актуальна по причине:

1. Java — язык программирования общего назначения. Имеет си-подобный синтаксис. Используется для создания приложений в разных областях: от веба до разработки игр, от мобильного ПО до программ для корпораций и научных институтов.
2. Java — язык программирования общего назначения. Относится к объектно-ориентированным языкам программирования, к языкам с сильной типизацией.
3. "Java является развивающимся высокоуровневым объектно-ориентированным языком программирования, который, к тому же является кросс-платформенным". Изучение этого языка программирования поможет в дальнейшем саморазвитии.

**Цель проекта.** Целью работы является закрепление знаний, полученных в процессе создания проекта, приобретение необходимых практических навыков для их дальнейшего применения.

В соответствие с целью были рассмотрены такие **задачи**:

1. Создание примитивного калькулятора с работающими вычислительными функциями.
2. Проанализировать работу и формулы системы счисления, с последующим добавлением их в программу.

**Глава 1. Система счислсния.**

* 1. **Краткая информация о системе.**

Изобретение позиционной нумерации, основанной на поместном значении цифр, приписывается шумерам и вавилонянам. В более поздний период такая нумерация была развита индусами и имела неоценимые последствия в истории цивилизации. К числу таких систем относится десятичная система счисления, возникновение которой связано со счётом на пальцах. В средневековой Европе она появилась через итальянских купцов, в свою очередь заимствовавших её у арабов.

**Позиционная система счисления** — система счисления, в которой значение цифры в записи числа зависит от её позиции в числе.

**Число**— это абсолютное понятие, выражающее количество. Одно и то же число может иметь различную*запись* в разных системах счисления.

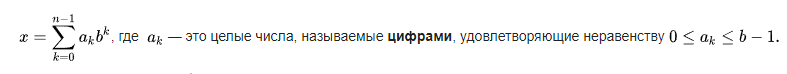
**Цифра**— это символ, используемый для записи чисел.

**Разряд**— это положение цифры в числе.

Позиционная система счисления определяется целым числом ***b>1***, нназываемым **основанием** системы счисления. Система счисления с основанием ***b*** так же называется ***b****-****ичной*** (в частности, двоичной, десятичной и т.п).

Количество цифр, используемых в системе счисления, определяется её основанием. В десятичной системе цифр десять, в двоичной системе — две.

Целое число без знака ***x***в***b****-ичной* системе счисления представляется в виде конечной линейной комбинации степеней числа ***b***.



Каждый базисный элемент ***b^k*** в таком представлении называется разрядом(позицией), старшинство разрядов и соответствующих им цифр определяется номером разряда (позиции) ***k*** (значением показателя степени)

С помощью ***n*** позиций в ***b-ичной*** системе счисления можно записать целые числа в диапазоне от ***0*** до ***b^(n-1),*** т. е. всего ***b^n*** различных чисел.

**Глава 2. Описание работы кода.**

* 1. **Основные свойства, формулы, используемые при счислении.**

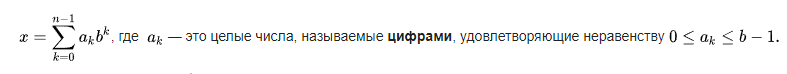
Позиционная система счисления обладает рядом свойств:

1. Основание системы счисления в ней самой всегда записывается как 10; например, в двоичной системе счисления 10 означает число 2. Данное утверждение неприменимо к унарной системе счисления, в которой используется только одна цифра.
2. Сравнивать числа, записанные в позиционной системе счисления, можно поразрядно, предварительно дополнив их ведущими нулями до одинаковой длины. При этом сравнение идёт от старшего разряда к младшему до тех пор, пока цифра в одном числе не будет больше соответствующей цифры в другом. Например, для сравнения чисел 321 и 312 в десятичной системе счисления сравниваются цифры в одинаковых разрядах слева направо:

* 3 = 3 — результат сравнения чисел на данном шаге не определён;
* 2 > 1 — первое число больше (независимо от оставшихся цифр).

Таким образом, естественный порядок на числах соответствует лексикографическому порядку на их записях в позиционной системе счисления при условии, что эти записи дополнены ведущими нулями до одинаковой длины.

1. Арифметические операции над числами. Позиционная система счисления позволяет без труда выполнять сложение, вычитание, умножение, деление и деление с остатком чисел, зная только таблицу сложения однозначных чисел, а для трёх последних операций ещё и таблицу умножения в соответствующей системе (см., например, деление столбиком).



**2.2 Разбор кода по частям.**

public void digitClick(ActionEvent actionEvent) {  
 String num = ((Button)actionEvent.getSource()).getText();  
  
 if (check) { // проверка числа  
 firstNumber = firstNumber + num;  
 } else {  
 currentNumber = currentNumber + num;  
 }  
 showText(actionEvent);  
}

**Функция прописывания цифр**

public void operationClick(ActionEvent actionEvent) {  
 check = false;  
 showText(actionEvent);  
 if (iterationNum > 1) {  
 solution();  
 }  
 currentNumber = "";  
 Calculation = ((Button)actionEvent.getSource()).getText();  
 iterationNum++;  
}

**Функция для простых арифметических действий**  
public void clearClick(ActionEvent actionEvent) {  
 textField.setText("");  
 txtHex.setText("");  
 txtDec.setText("");  
 txtOct.setText("");  
 txtBin.setText("");  
 check = true;  
 currentNumber = "";  
 firstNumber = "";  
 iterationNum = 1;  
}

**Функция для очистки**  
  
public void showText(ActionEvent actionEvent) {  
 textField.setText(textField.getText() + ((Button)actionEvent.getSource()).getText());  
}

**Функция для показа операция на экране**  
public void solution() {  
 int first = Integer.*parseInt*(firstNumber);  
 int second = Integer.*parseInt*(currentNumber);  
 int res;  
 switch (Calculation) {  
 case "\*":  
 res = first \* second;  
 firstNumber = "" + res;  
 break;  
 case "/":  
 res = first / second;  
 firstNumber = "" + res;  
 break;  
 case "+":  
 res = first + second;  
 firstNumber = "" + res;  
 break;  
 case "-":  
 res = first - second;  
 firstNumber = "" + res;  
 break;  
 }  
}  
**Функция для решения арифметических действий**

@FXML  
void equalClick(ActionEvent actionEvent) {  
 solution();  
 textField.setText("");  
 check = false;  
 textField.setText(firstNumber);  
 iterationNum = 1;  
}  
**Функция расчета**

public void letClick(ActionEvent actionEvent) {  
 String num = ((Button)actionEvent.getSource()).getText();  
  
 if (check) { // проверка  
 firstNumber = firstNumber + num;  
 } else {  
 currentNumber = currentNumber + num;  
 }  
 showText(actionEvent);  
}

**Функция прописывания букв**

public void onHexClick(ActionEvent actionEvent) {  
 btn0.setDisable(false);  
 btn1.setDisable(false);  
 btn2.setDisable(false);  
 btn3.setDisable(false);  
 btn4.setDisable(false);  
 btn5.setDisable(false);  
 btn6.setDisable(false);  
 btn7.setDisable(false);  
 btn8.setDisable(false);  
 btn9.setDisable(false);  
 btnA.setDisable(false);  
 btnB.setDisable(false);  
 btnC.setDisable(false);  
 btnD.setDisable(false);  
 btnE.setDisable(false);  
 btnF.setDisable(false);  
}

public void onOctClick(ActionEvent actionEvent) {  
 btn0.setDisable(false);  
 btn1.setDisable(false);  
 btn2.setDisable(false);  
 btn3.setDisable(false);  
 btn4.setDisable(false);  
 btn5.setDisable(false);  
 btn6.setDisable(false);  
 btn7.setDisable(false);  
 btn8.setDisable(true);  
 btn9.setDisable(true);  
 btnA.setDisable(true);  
 btnB.setDisable(true);  
 btnC.setDisable(true);  
 btnD.setDisable(true);  
 btnE.setDisable(true);  
 btnF.setDisable(true);  
}

public void onDecClick(ActionEvent actionEvent) {  
 btn0.setDisable(false);  
 btn1.setDisable(false);  
 btn2.setDisable(false);  
 btn3.setDisable(false);  
 btn4.setDisable(false);  
 btn5.setDisable(false);  
 btn6.setDisable(false);  
 btn7.setDisable(false);  
 btn8.setDisable(false);  
 btn9.setDisable(false);  
 btnA.setDisable(true);  
 btnB.setDisable(true);  
 btnC.setDisable(true);  
 btnD.setDisable(true);  
 btnE.setDisable(true);  
 btnF.setDisable(true);  
}  
  
public void onBinClick(ActionEvent actionEvent) {  
 btn0.setDisable(false);  
 btn1.setDisable(false);  
 btn2.setDisable(true);  
 btn3.setDisable(true);  
 btn4.setDisable(true);  
 btn5.setDisable(true);  
 btn6.setDisable(true);  
 btn7.setDisable(true);  
 btn8.setDisable(true);  
 btn9.setDisable(true);  
 btnA.setDisable(true);  
 btnB.setDisable(true);  
 btnC.setDisable(true);  
 btnD.setDisable(true);  
 btnE.setDisable(true);  
 btnF.setDisable(true);  
}  
**Функции для работы с системой счисления**

public void toDec() {  
 int cc = Integer.*parseInt*(firstNumber);  
 txtBin.setText(Integer.*toBinaryString*(cc));  
 txtOct.setText(Integer.*toOctalString*(cc));  
 txtHex.setText(Integer.*toHexString*(cc));  
 txtDec.setText(textField.getText());  
}  
public void toBin() {  
 int cc = Integer.*parseInt*(firstNumber);  
 String as = textField.getText();  
 txtBin.setText(textField.getText());  
 txtOct.setText(Integer.*toOctalString*(Integer.*parseInt*(as, 2)));  
 txtDec.setText(String.*valueOf*(Integer.*parseInt*(as, 2)));  
 txtHex.setText(Integer.*toHexString*(Integer.*parseInt*(as, 2)));  
}  
public void toHex() {  
 String as = textField.getText();  
 txtBin.setText(Integer.*toBinaryString*(Integer.*parseInt*(as, 16)));  
 txtOct.setText(Integer.*toOctalString*(Integer.*parseInt*(as, 16)));  
 txtDec.setText(String.*valueOf*(Integer.*parseInt*(as, 16)));  
 txtHex.setText(textField.getText());  
}

public void toOct() {  
 int cc = Integer.*parseInt*(firstNumber);  
 String as = textField.getText();  
 txtBin.setText(Integer.*toBinaryString*(Integer.*parseInt*(as, 8)));  
 txtOct.setText(textField.getText());  
 txtDec.setText(String.*valueOf*(Integer.*parseInt*(as, 8)));  
 txtHex.setText(Integer.*toHexString*(Integer.*parseInt*(as, 8)));  
  
}

**Функции расчета системы счислении**

public void decClick(ActionEvent actionEvent) {  
 toDec();  
}  
public void binClick(ActionEvent actionEvent) {  
 toBin();  
}  
public void hexClick(ActionEvent actionEvent) {  
 toHex();  
}  
public void octClick(ActionEvent actionEvent) {  
 toOct();  
}

**Функции запуска расчета**

Я прописал и описал основные части моего кода, остальное можно посмотреть уже в самом коде.

**Заключение**

В заключении, хочу написать, что все поставленные выше задачи были выполнены. В завершение получился код, облегчающий как программистам, так и обычным людям использующим калькулятор вычисление примитивных чисел, а так же перевод в системе счисления. За основу были взяты функции шифратора и дешифратора.

В ходе выполнения работы были более подробно изучены и изложены вопросы:

- Основная информация по языку Java.

- Ключевые алгоритмы и фрагменты программы, отвечающие за корректную работу программы.

Результатом работы является программистский калькулятор.

**Список литературы**

1. <https://foxford.ru/wiki/informatika/pozitsionnye-sistemy-schisleniya>
2. <https://ru.wikibooks.org/wiki/Системы_счисления>
3. <https://metanit.com/java/tutorial/1.1.php>
4. <https://ru.wikipedia.org/w/index.php?title=Позиционная_система_счисления&stable=1>