

Новосибирск, 2023
ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАТИКИ»

Кафедра вычислительных систем

КУРСОВАЯ РАБОТА
по дисциплине «Технологии разработки программного обеспечения»
на тему «Units convertor»

Выполнил:
ст. гр. ИС-241
Сотников Павел
Проверил:
ст. преподаватель Токмашева Е. И.

<u>Введение и постановка задачи</u>	2
<u>Техническое задание</u>	3
<u>Описание выполненного проекта</u>	4
<u>Личный вклад в проект</u>	7
<u>Приложение. Текст программы</u>	8

Введение и постановка задачи

Цель работы: создать приложение, конвертирующее единицы измерения (Units Converter).

Постановка задачи: для разработки был выбран язык программирования Си. В своей программе мы решили реализовать переводы таких величин измерения как: длины (сантиметры, дециметры, метры километры)., температуры (цельсии, фаренгейты, кельвины), времени (секунды, минуты, часы, дни, месяцы, года).

Техническое задание

Наш продукт решает задачу конвертации разного рода величин. Сначала, пользователем вводится число и величина, которые следует конвертировать. Затем, на выбор дается ряд доступных для перевода величин. После завершения работы программы в терминал выводится готовый ответ.

Описание выполненного проекта

При запуске программы на экране выводится приветствие и все доступные величины для перевода.

```
## ## ## ## ##### #####
## ## ### ## ## ##
## ## ## ### ## ##
## ## ## ## ## ##
#### ## ## ##### ##

#### #### ## ## ## ## ##### ## ## ##### ##### #####
## ## ## ## ### ## ## ## ## ## ## ## ## ## ## ##
## ## ## ## ## ### ## ## ##### # ### ## ##### #####
## # ## ## ## ## ## ##### ## ## ## ## ## ## ##
#### #### ## ## ## ##### ## ## ## ##### ## ##
```

----- ----- ----- ----- -----
----- ----- CONVERSION IN ----- -----
Millimeters Centimeters Decimeters Meters Kilometers
----- ----- ----- ----- -----
----- ----- CONVERSION TO ----- -----
Centimeters Milimeters Milimeters Milimeters Milimeters
Decimeters Decimeters Centimeters Centimeters Centimeters
Meters Meters Meters Decimeters Decimeters
Kilometers Kilometers Kilometers Kilometers Meters
----- ----- ----- ----- -----

----- ----- -----
----- CONVERSION IN -----
Celsius Fahrenheit Kelvin
----- ----- -----
----- CONVERSION TO -----
Fahrenheit Kelvin Celsius
Kelvin Fahrenheit Celsius
----- ----- -----

	CONVERSION IN	
Grams	Kilograms	Hundredweight
	CONVERSION TO	
Kilograms	Grams	Grams
Hundredweight	Hundredweight	Kilograms

		CONVERSION IN			
Seconds	Minutes	Hours	Days	Months	Years
		CONVERSION TO			
Minutess	Seconds	Seconds	Seconds	Seconds	Seconds
Hours	Hours	Minutes	Minutes	Minutes	Minutes
Days	Days	Days	Hours	Hours	Hours
Months	Months	Months	Months	Days	Days
Years	Years	Years	Years	Years	Months
<div> <div>+</div> <div>-----+</div> <div>Enter a value: </div> <div></div> </div>					

Пользователь вводит значение и единицу измерения, которую он хочет конвертировать. Затем ему на выбор дается доступные единицы измерения, в которые он хочет конвертировать.

```

+-----+
| Enter a value:|      2 |
+-----+

+-----+
| Enter a unit:|     cm |
+-----+

| Choose conversion: |
+-----+
| Option |      | Conversion |
+-----+
| a) |      | CM to M |
+-----+
| b) |      | CM to MM |
+-----+
| c) |      | CM to KM |
+-----+
| d) |      | CM to DC |
+-----+

+-----+
| Enter a choise:|     a |
+-----+

+-----+
| Your answer is: 2.00 centimeters is equal to 0.02 meters.
+-----+

```

На данном скриншоте показана работа программы. Сначала мы вводим число два, потом уточняем, что это сантиметры.

Следующий шаг — это выбор в какую единицу длины пользователь хотел бы перевести два сантиметра. Пользователь выбирает вариант под номером А, получает готовый ответ, и работа программы завершается.

Личный вклад в проект.

Личный вклад я могу оценить в 91%, потому что почти все, что связано с кодом разбирался и писал лично я. За исключением презентации и отчета. Надеюсь, вы заметите это.

Приложение. Текст программы

Coursework.c

```
// реализация наших функций
#include <stdio.h>
#include "coursework.h"

float result;
// Функция для конвертации САНТИМЕТРОВ
int convertSM(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value / 100;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f centimeters is equal to %.2f meters. \n", value,
result);
            printf("+-----+\n");
            return result;

        case 'b':
            result = value * 10;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f centimeters is equal to %.2f millimeters. \n",
value, result);
            printf("+-----+\n");
            return result;

        case 'c':
            result = value / 100000;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f centimeters is equal to %.5f kilometers. \n",
value, result);
            printf("+-----+\n");
            return result;

        case 'd':
            result = value / 10;
            printf("+-----+\n");
            printf("| Your answer is: %.2f centimeters is equal to %.2f decimeter. \n",
value, result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}
// Функция для конвертации МЕТРОВ
```

```

int convertationM(float value, char choise)
{
    switch (choise)
    {
        case 'a':
            result = value * 100;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f meters is equal to %.2f centimeters. \n", value,
result);
            printf("+-----+\n");
            return result;

        case 'b':
            result = value / 1000;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f meters is equal to %.3f kilometers. \n", value,
result);
            printf("+-----+\n");
            return result;

        case 'c':
            result = value / 1069;
            printf("+-----+\n");
            printf("| Your answer is: %.2f meters is equal to %.5f miles. \n", value,
result);
            printf("+-----+\n");
            return result;

        case 'd':
            result = value * 3.281;
            printf("+-----+\n");
            printf("| Your answer is: %.2f meters is equal to %.2f foot. \n", value,
result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}

// функция для конвертации МИЛЛИМЕТРОВ
int convertMM(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value / 10;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Millimeters is equal to %.2f Centimeters. \n",
value, result);
            printf("+-----+\n");
            return result;
    }
}

```

```

        case 'b':
            result = value / 1000;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Millimeters is equal to %.2f Meters. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'c':
            result = value * 1e+6;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Millimeters is equal to %.2f Kilometers. \n",
value, result);
            printf("+-----+\n");
            return result;
        case 'd':
            result = value / 304.8;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Millimeters is equal to %.5f Pouns. \n", value,
result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}
// функция для конвертации ЦЕЛЬСИЯ
int convertC(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = (value * 9 / 5) + 32;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Celsius is equal to %.2f Fahrenheit. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = value + 273;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Celsius is equal to %.2f Kelvins. \n", value,
result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}

```

```

}
// функция для конвертации КЕЛЬВИНЫ
int convertK(float value, char choise)
{
    switch (choise)
    {
        case 'a':
            result = 1.8 * (value - 273) + 32;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f kelvin is equal to %.2f Fahrenheit. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = value - 273;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f kelvin is equal to %.2f Celsius. \n", value,
result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}

// функция для ФАРЕНГЕЙТ
int convertF(float value, char choise)
{
    switch (choise)
    {
        case 'a':
            result = (5.0 / 9.0) * (value - 32) + 273.15;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Fahrenheit is equal to %.2f Kelvin. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = (value - 32) * (5.0 / 9.0);
            printf("+-----+\n");
            printf("| Your answer is: %.2f Fahrenheit is equal to %.2f Celsius. \n", value,
result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}
// функция для КИЛОГРАММОВ

```

```

int convertKG(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value * 1000;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Kilograms is equal to %.2f Grams. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = value * 2.205;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Kilograms is equal to %.2f Pounds. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'c':
            result = value * 1e+6;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Kilograms is equal to %.2f Miligrams. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'd':
            result = value * 5000;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Kilograms is equal to %.2f Carrats. \n", value,
result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}

// функция для ГРАММОВ
int convertGR(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value / 1000;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Grams is equal to %.3f Kilograms. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':

```

```

        result = value * 1000;
        printf("\n\n");
        printf("+-----+\n");
        printf("| Your answer is: %.2f Grams is equal to %.2f Milligrams. \n", value,
result);
        printf("+-----+\n");
        return result;
    case 'c':
        result = value / 453.6;
        printf("\n\n");
        printf("+-----+\n");
        printf("| Your answer is: %.2f Grams is equal to %.4f Pounds. \n", value,
result);
        printf("+-----+\n");
        return result;
    case 'd':
        result = value * 5;
        printf("\n\n");
        printf("+-----+\n");
        printf("| Your answer is: %.2f Grams is equal to %.2f Carrats. \n", value,
result);
        printf("+-----+\n");
        return result;
    case 'e':
        result = value / 1e+6;
        printf("\n\n");
        printf("+-----+\n");
        printf("| Your answer is: %.2f Grams is equal to %.6f Tonns. \n", value,
result);
        printf("+-----+\n");
        return result;

    default:
        printf("| Invalid choice. |\n");
        break;
}
}
// функция для ЦЕНТНЕРОВ
int convertHW(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value * 100000;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Hundredweight is equal to %.2f Grams. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = value * 100;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Hundredweight is equal to %.2f Kilograms. \n",
value, result);

```

```

        printf("+-----+\n");
        return result;
    case 'c':
        result = value / 10;
        printf("\n\n");
        printf("+-----+\n");
        printf("| Your answer is: %.2f Hundredweight is equal to %.2f Tons. \n", value,
result);
        printf("+-----+\n");
        return result;
    case 'd':
        result = value * 220.5;
        printf("\n\n");
        printf("+-----+\n");
        printf("| Your answer is: %.2f Hundredweight is equal to %.2f Pounds. \n",
value, result);
        printf("+-----+\n");
        return result;

    default:
        printf("| Invalid choice. |\n");
        break;
    }
}
// функция для СЕКУНД
int convertSEC(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value / 60;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Seconds is equal to %.3f Minutes. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = value / 3600;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Seconds is equal to %.5f Hours. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'c':
            result = value / 86400;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Seconds is equal to %.6f Days. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'd':
            result = value / 2592000;
            printf("\n\n");

```

```

        printf("+-----+\n");
        printf("| Your answer is: %.2f Seconds is equal to %.9f Months. \n", value,
result);
        printf("+-----+\n");
        return result;
    case 'e':
        result = value / 31536000;
        printf("\n\n");
        printf("+-----+\n");
        printf("| Your answer is: %.2f Seconds is equal to %.10f Years. \n", value,
result);
        printf("+-----+\n");
        return result;

    default:
        printf("| Invalid choice. |\n");
        break;
    }
}
// функция для МИНУТ
int convertMIN(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value * 60;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Minutes is equal to %.2f Seconds. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = value / 60;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Minutes is equal to %.3f Hours. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'c':
            result = value / 1440;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Minutes is equal to %.4f Days. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'd':
            result = value / 43200;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Minutes is equal to %.6f Months. \n", value,
result);
            printf("+-----+\n");
            return result;
    }
}

```



```

        case 'e':
            result = value / 525600;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Minutes is equal to %.7f Years. \n", value,
result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}
// функция для ЧАСОВ
int convertHOUR(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value * 3600;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Hours is equal to %.2f Seconds. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = value * 60;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Hours is equal to %.2f Minutes. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'c':
            result = value / 24;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Hours is equal to %.2f Days. \n", value, result);
            printf("+-----+\n");
            return result;
        case 'd':
            result = value / 730;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Hours is equal to %.3f Months. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'e':
            result = value / 8760;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Hours is equal to %.4f Years. \n", value,
result);

```

```

        printf("+-----+\n");
        return result;

    default:
        printf("| Invalid choice. |\n");
        break;
    }
}
// функция для ДНЕЙ
int convertDAY(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value * 86400;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Days is equal to %.2f Seconds. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = value * 1440;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Days is equal to %.2f Minutes. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'c':
            result = value * 24;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Days is equal to %.2f Hours. \n", value, result);
            printf("+-----+\n");
            return result;
        case 'd':
            result = value / 30.417;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Days is equal to %.3f Months. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'e':
            result = value / 365;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Days is equal to %.4f Years. \n", value, result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}

```

```

}
// функция для конвертации МЕСЯЦА
int convertMONTH(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value * 2.628e+6;
            printf("\n\n");
            printf("+-----+\n");
            printf("| Your answer is: %.2f Months is equal to %.7f Seconds. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'b':
            result = value * 43800;
            printf("+-----+\n");
            printf("| Your answer is: %.2f Months is equal to %.6f Minutes. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'c':
            result = value * 730;
            printf("+-----+\n");
            printf("| Your answer is: %.2f Months is equal to %.4f Hours. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'd':
            result = value * 30.417;
            printf("+-----+\n");
            printf("| Your answer is: %.2f Months is equal to %.2f Days. \n", value,
result);
            printf("+-----+\n");
            return result;
        case 'e':
            result = value / 12;
            printf("+-----+\n");
            printf("| Your answer is: %.2f Months is equal to %.2f Years. \n", value,
result);
            printf("+-----+\n");
            return result;

        default:
            printf("| Invalid choice. |\n");
            break;
    }
}
// функция для конвертации года
int convertYEAR(float value, char choice)
{
    switch (choice)
    {
        case 'a':
            result = value * 3.154e+7;
            printf("\n\n");

```

```

        printf("+-----+\n");
        printf("| Your answer is: %.2f Years is equal to %.8f Seconds. \n", value,
result);
        printf("+-----+\n");
        return result;
    case 'b':
        result = value * 525600;
        printf("+-----+\n");
        printf("| Your answer is: %.2f Years is equal to %.7f Minutes. \n", value,
result);
        printf("+-----+\n");
        return result;
    case 'c':
        result = value * 8760;
        printf("+-----+\n");
        printf("| Your answer is: %.2f Years is equal to %.4f Hours. \n", value,
result);
        printf("+-----+\n");
        return result;
    case 'd':
        result = value * 365;
        printf("+-----+\n");
        printf("| Your answer is: %.2f Years is equal to %.3f Days. \n", value, result);
        printf("+-----+\n");
        return result;
    case 'e':
        result = value * 12;
        printf("+-----+\n");
        printf("| Your answer is: %.2f Years is equal to %.2f Months. \n", value,
result);
        printf("+-----+\n");
        return result;

    default:
        printf("| Invalid choice. |\n");
        break;
}
}

```

Coursework.h

```

// инициализация наших реализованных функций
#include <stdio.h>
int convertSM(float value, char choice);
int convertationM(float value, char choice);
int convertMM(float value, char choice);
int convertC(float value, char choice);
int convertK(float value, char choice);
int convertF(float value, char choice);
int convertKG(float value, char choice);
int convertGR(float value, char choice);
int convertHW(float value, char choice);

```

```
int convertSEC(float value, char choice);
int convertMIN(float value, char choice);
int convertHOUR(float value, char choice);
int convertDAY(float value, char choice);
int convertMONTH(float value, char choice);
int convertYEAR(float value, char choice);
```

Main.c

[illegible]

```

printf("| %10s | %10s | %15s | %10s | %10s |\n", "Milimeters", "Centimeters", "Decimeters",
"Meters", "Kilometers");
printf("-----|-----|-----|-----|-----|\n");
printf("|-----|-----| CONVERSION TO |-----|-----|\n");
printf("-----|-----|-----|-----|-----|\n");
printf("| %10s | %10s | %15s | %10s | %10s | \n", "Centimeters", "Milimeters",
"Milimeters", "Milimeters", "Milimeters");
printf("-----|-----|-----|-----|-----|\n");
printf("| %10s | %10s | %15s | %10s | %10s |\n", "Decimeters", "Decimeters", "Centimeters",
"Centimeters", "Centimeters");
printf("-----|-----|-----|-----|-----|\n");
printf("| %10s | %10s | %15s | %10s | %10s | \n", "Meters", "Meters", "Meters",
"Decimeters", "Decimeters");
printf("-----|-----|-----|-----|-----|\n");
printf("| %10s | %10s | %15s | %10s | %10s | \n", "Kilometers", "Kilometers", "Kilometers",
"Kilometers", "Meters");
printf("-----|-----|-----|-----|-----|\n");
printf("\n\n\n");

printf("-----|-----|-----|\n");
printf("|-----| CONVERSION IN |-----|\n");
printf("-----|-----|-----|\n");
printf("| %6s | %10s | %6s | \n", "Celsius", "Fahrenheit", "Kelvin");
printf("-----|-----|-----|\n");
printf("|-----| CONVERSION TO |-----|\n");
printf("-----|-----|-----|\n");
printf("| %6s | %6s | %6s | \n", "Fahrenheit", "Kelvin", "Celsius");
printf("-----|-----|-----|\n");
printf("| %6s | %6s | %6s | \n", "Kelvin", "Fahrenheit", "Celsius");
printf("-----|-----|-----|\n");
printf("\n\n\n");

printf("-----|-----|-----|\n");
printf("|-----| CONVERSION IN |-----|\n");
printf("-----|-----|-----|\n");
printf("| %6s | %10s | %6s | \n", "Grams", "Kilograms", "Hundredweight");
printf("-----|-----|-----|\n");
printf("|-----| CONVERSION TO |-----|\n");
printf("-----|-----|-----|\n");
printf("| %6s | %6s | %6s | \n", "Kilograms", "Grams", "Grams");
printf("-----|-----|-----|\n");
printf("| %6s | %6s | %6s | \n", "Hundredweight", "Hundredweight", "Kilograms");
printf("-----|-----|-----|\n");
printf("\n\n\n");

printf("\n\n\n");
printf("-----|-----|-----|-----|-----|-----|
-|\n");
printf("|-----|-----| CONVERSION IN |-----|-----|-----|
-|\n");
printf("-----|-----|-----|-----|-----|-----|
-|\n");
printf("| %10s | %10s | %15s | %10s | %10s | %10s | \n", "Seconds", "Minutes", "Hours",
"Days", "Months", "Years");
printf("-----|-----|-----|-----|-----|-----|
-|\n");

```

```

printf("|-----|-----|  CONVERSION TO  |-----|-----|-----|
-|\n");
printf("-----|-----|-----|-----|-----|-----|
-|\n");
printf("| %10s | %10s | %15s | %10s | %10s | %10s | \n", "Minutess", "Seconds", "Seconds",
"Seconds", "Seconds", "Seconds");
printf("-----|-----|-----|-----|-----|-----|
-|\n");
printf("| %10s | %10s | %15s | %10s | %10s | %10s | \n", "Hours", "Hours", "Minutes",
"Minutes", "Minutes", "Minutes");
printf("-----|-----|-----|-----|-----|-----|
-|\n");
printf("| %10s | %10s | %15s | %10s | %10s | %10s | \n", "Days", "Days", "Days", "Hours",
"Hours", "Hours");
printf("-----|-----|-----|-----|-----|-----|
-|\n");
printf("| %10s | %10s | %15s | %10s | %10s | %10s | \n", "Months", "Months", "Months",
"Months", "Days", "Days");
printf("-----|-----|-----|-----|-----|-----|
-|\n");
printf("| %10s | %10s | %15s | %10s | %10s | %10s | \n", "Years", "Years", "Years",
"Years", "Years", "Months");
printf("-----|-----|-----|-----|-----|-----|
-|\n");
printf("\n\n\n");

printf("+-----+\n");
printf("| Enter a value:| ");
scanf("%f", &value);
printf("+-----+\n");
printf("\n\n");

if (value != (int)value)
{
printf("\033[0;31m"); // установка красного цвета
printf("+-----+\n");
printf("|input error: not a numeric value|\n");
printf("+-----+\n");
printf("\033[0m"); // убираем
return 0;
}
printf("\033[0;37;1m"); // установка цвета (белый)

printf("+-----+\n");
printf("| Enter a unit:| ");
scanf(" %s", units);
printf("+-----+\n");
printf("\n\n");

int P = 0;
while (units[P] != '\0')
{
if (isdigit(units[P]))
{
printf("\033[0;31m"); // установка красного цвета
printf("+-----+\n");

```

```

        printf("|input error: incorrect value entered|\n");
        printf("+-----+\n");
        printf("\033[0m"); // убираем
        return 1;
    }
    P++;
}
printf("\033[0;37;1m"); // установка цвета (белый)

int isValidUnit = 0;
while (!isValidUnit)
{
    // определяем, какую единицу измерения ввели с клавиатуры(В ДАННОМ СЛУЧАЕ САНТИМЕТРЫ!!)
    if (strcmp(units, "cm") == 0)
    {

        printf("|    Choose conversion:    |\n");
        printf("+-----+\n");
        printf("| Option |    | Conversion |\n");
        printf("+-----+\n");
        printf("| a) |    | CM to M |\n");
        printf("+-----+\n");
        printf("| b) |    | CM to MM |\n");
        printf("+-----+\n");
        printf("| c) |    | CM to KM |\n");
        printf("+-----+\n");
        printf("| d) |    | CM to DC |\n");
        printf("+-----+\n");

        printf("\n\n");
        printf("+-----+\n");
        printf("| Enter a choise:|    |");
        scanf(" %c", &choice);
        printf("+-----+\n");
        convertSM(value, choice);
        printf("\n\n");
        isValidUnit = 1;
    }

    // определяем, какую единицу измерения ввели с клавиатуры(В ДАННОМ СЛУЧАЕ МЕТРЫ!!)
    else if (strcmp(units, "m") == 0)
    {
        printf("|    Choose conversion:    |\n");
        printf("+-----+\n");
        printf("| Option |    | Conversion |\n");
        printf("+-----+\n");
        printf("| a) |    | M to CM |\n");
        printf("+-----+\n");
        printf("| b) |    | M to KM |\n");
        printf("+-----+\n");
        printf("| c) |    | M to Ml |\n");
        printf("+-----+\n");
        printf("| d) |    | M to Ft |\n");
        printf("+-----+\n");
        printf("\n\n");
        printf("+-----+\n");
    }
}

```



```

printf("| Enter a choose:|      ");
scanf(" %c", &choice);
printf("+-----+\n");
conversionM(value, choice);
printf("\n\n");
isValidUnit = 1;
}
// определяем, какую единицу измерения ввели с клавиатуры (В ДАННОМ СЛУЧАЕ МИЛЛИМЕТРЫ)
else if (strcmp(units, "mm") == 0)
{
printf("| Choose conversion: | \n");
printf("+-----+\n");
printf("| a) |      | MM to CM | \n");
printf("+-----+\n");
printf("| b) |      | MM to M  | \n");
printf("+-----+\n");
printf("| c) |      | MM to KM | \n");
printf("+-----+\n");
printf("| d) |      | MM to PD | \n");
printf("+-----+\n");

printf("+-----+\n");
printf("| Enter a choose:|      ");
scanf(" %c", &choice);
printf("+-----+\n");
convertMM(value, choice);
isValidUnit = 1;
}
// определяем, какую единицу измерения ввели с клавиатуры (В ДАННОМ СЛУЧАЕ ЦЕЛЬСИЯ)
else if (strcmp(units, "c") == 0)
{
printf("| Choose conversion: | \n");
printf("+-----+\n");
printf("| a) |      | C to F  | \n");
printf("+-----+\n");
printf("| b) |      | C to K  | \n");
printf("+-----+\n");

printf("+-----+\n");
printf("| Enter a choose:|      ");
scanf(" %c", &choice);
printf("+-----+\n");
convertC(value, choice);
isValidUnit = 1;
}
// определяем, какую единицу измерения ввели с клавиатуры (В ДАННОМ СЛУЧАЕ КЕЛЬВИНЫ)
else if (strcmp(units, "k") == 0)
{
printf("| Choose conversion: | \n");
printf("+-----+\n");
printf("| a) |      | K to F  | \n");
printf("+-----+\n");
printf("| b) |      | K to C  | \n");
printf("+-----+\n");

printf("+-----+\n");

```

```

printf("| Enter a choose:|      ");
scanf(" %c", &choice);
printf("+-----+\n");
convertK(value, choice);
isValidUnit = 1;
}

// определяем, какую единицу измерения ввели с клавиатуры (В ДАННОМ СЛУЧАЕ ФАРЕНГЕЙТЫ)
else if (strcmp(units, "f") == 0)
{
    printf("| Choose conversion: | \n");
    printf("+-----+\n");
    printf("| a) | | F to K | \n");
    printf("+-----+\n");
    printf("| b) | | F to C | \n");
    printf("+-----+\n");

    printf("+-----+\n");
    printf("| Enter a choose:|      ");
    scanf(" %c", &choice);
    printf("+-----+\n");
    convertF(value, choice);
    isValidUnit = 1;
}

// определяем, какую единицу измерения ввели с клавиатуры (В ДАННОМ СЛУЧАЕ КИЛОГРАММЫ)
else if (strcmp(units, "kg") == 0)
{
    printf("| Choose conversion: | \n");
    printf("+-----+\n");
    printf("| a) | | KG to GR | \n");
    printf("+-----+\n");
    printf("| b) | | KG to PD | \n");
    printf("+-----+\n");
    printf("| c) | | KG to MG | \n");
    printf("+-----+\n");
    printf("| d) | | KG to CR | \n");
    printf("+-----+\n");

    printf("+-----+\n");
    printf("| Enter a choose:|      ");
    scanf(" %c", &choice);
    printf("+-----+\n");
    convertKG(value, choice);
    isValidUnit = 1;
}

// определяем, какую единицу измерения ввели с клавиатуры (В ДАННОМ СЛУЧАЕ ГРАММЫ)
else if (strcmp(units, "gr") == 0)
{
    printf("| Choose conversion: | \n");
    printf("+-----+\n");
    printf("| a) | | GR to KG | \n");
    printf("+-----+\n");
    printf("| b) | | GR to MG | \n");
    printf("+-----+\n");
    printf("| b) | | GR to PD | \n");

```

```

printf("+-----+\\n");
printf("| d) |          | GR to CR |\\n");
printf("+-----+\\n");
printf("| e) |          | GR to TN |\\n");
printf("+-----+\\n");

printf("+-----+\\n");
printf("| Enter a choice:|          |");
scanf(" %c", &choice);
printf("+-----+\\n");
convertGR(value, choice);
isValidUnit = 1;
}

// определяем, какую единицу измерения ввели с клавиатуры ( В ДАННОМ СЛУЧАЕ ЦЕНТНЕРЫ)
else if (strcmp(units, "hw") == 0)
{
    printf("| Choose conversion: |\\n");
    printf("+-----+\\n");
    printf("| a) |          | HW to GR |\\n");
    printf("+-----+\\n");
    printf("| b) |          | HW to KG |\\n");
    printf("+-----+\\n");
    printf("| c) |          | HW to TN |\\n");
    printf("+-----+\\n");
    printf("| d) |          | HW to PD |\\n");
    printf("+-----+\\n");

    printf("+-----+\\n");
    printf("| Enter a choice:|          |");
    scanf(" %c", &choice);
    printf("+-----+\\n");
    convertHW(value, choice);
    isValidUnit = 1;
}

// определяем, какую единицу измерения ввели с клавиатуры (В ДАННОМ СЛУЧАЕ СЕКУНДЫ)
else if (strcmp(units, "sec") == 0)
{
    printf("| Choose conversion: |\\n");
    printf("+-----+\\n");
    printf("| a) |          | SEC to MIN |\\n");
    printf("+-----+\\n");
    printf("| b) |          | SEC to HOUR |\\n");
    printf("+-----+\\n");
    printf("| c) |          | SEC to DAYS |\\n");
    printf("+-----+\\n");
    printf("| d) |          | SEC to MONTHS |\\n");
    printf("+-----+\\n");
    printf("| e) |          | SEC to YEAR |\\n");
    printf("+-----+\\n");

    printf("+-----+\\n");
    printf("| Enter a choice:|          |");
    scanf(" %c", &choice);
    printf("+-----+\\n");
    convertSEC(value, choice);
}

```

```

        isValidUnit = 1;
    }

    // определяем, какую единицу измерения ввели клавиатуры (в данном случае минуты)
    else if (strcmp(units, "min") == 0)
    {
        printf("| Choose conversion: |\n");
        printf("+-----+\n");
        printf("| a) | MIN to SEC |\n");
        printf("+-----+\n");
        printf("| b) | MIN to HOURS |\n");
        printf("+-----+\n");
        printf("| c) | MIN to DAYS |\n");
        printf("+-----+\n");
        printf("| d) | MIN to MONTHS |\n");
        printf("+-----+\n");
        printf("| e) | MIN to YEARS |\n");
        printf("+-----+\n");

        printf("+-----+\n");
        printf("| Enter a choice:| ");
        scanf(" %c", &choice);
        printf("+-----+\n");
        convertMIN(value, choice);
        isValidUnit = 1;
    }

    // определяем, какую единицу измерения ввели с клавиатуры (в данном случае часы)
    else if (strcmp(units, "hour") == 0)
    {
        printf("| Choose conversion: |\n");
        printf("+-----+\n");
        printf("| a) | HOURS to SEC |\n");
        printf("+-----+\n");
        printf("| b) | HOURS to MIN |\n");
        printf("+-----+\n");
        printf("| c) | HOURS to DAYS |\n");
        printf("+-----+\n");
        printf("| d) | HOURS to MONTHS |\n");
        printf("+-----+\n");
        printf("| e) | HOURS to YEARS |\n");
        printf("+-----+\n");

        printf("+-----+\n");
        printf("| Enter a choice:| ");
        scanf(" %c", &choice);
        printf("+-----+\n");
        convertHOUR(value, choice);
        isValidUnit = 1;
    }

    // определяем, какую единицу измерения ввели с клавиатуры (в данном случае дни)
    else if (strcmp(units, "day") == 0)
    {
        printf("| Choose conversion: |\n");
        printf("+-----+\n");
        printf("| a) | DAYS to SEC |\n");

```

```

printf("+-----+\n");
printf("| b) |          | DAYS to MIN  |\n");
printf("+-----+\n");
printf("| c) |          | DAYS to HOURS |\n");
printf("+-----+\n");
printf("| d) |          | DAYS to MONTHS |\n");
printf("+-----+\n");
printf("| e) |          | DAYS to YEARS |\n");
printf("+-----+\n");

printf("+-----+\n");
printf("| Enter a choose:|          |");
scanf(" %c", &choice);
printf("+-----+\n");
convertDAY(value, choice);
isValidUnit = 1;
}
// определяем, какую единицу измерения ввели с клавиатуры (в данном случае месяца)
else if (strcmp(units, "month") == 0)
{
    printf("| Choose conversion:      |\n");
    printf("+-----+\n");
    printf("| a) |          | MONTHS to SEC  |\n");
    printf("+-----+\n");
    printf("| b) |          | MONTHS to MIN  |\n");
    printf("+-----+\n");
    printf("| c) |          | MONTHS to HOURS |\n");
    printf("+-----+\n");
    printf("| c) |          | MONTHS to DAYS  |\n");
    printf("+-----+\n");
    printf("| e) |          | MONTHS to YEARS |\n");
    printf("+-----+\n");

    printf("+-----+\n");
    printf("| Enter a choose:|          |");
    scanf(" %c", &choice);
    printf("+-----+\n");
    convertMONTH(value, choice);
    isValidUnit = 1;
}
// определяем с клавиатуры, какую единицу измерения ввели (в данном случае года)
else if (strcmp(units, "year") == 0)
{
    printf("| Choose conversion:      |\n");
    printf("+-----+\n");
    printf("| a) |          | YEARS to SEC   |\n");
    printf("+-----+\n");
    printf("| b) |          | YEARS to MIN   |\n");
    printf("+-----+\n");
    printf("| c) |          | YEARS to HOURS |\n");
    printf("+-----+\n");
    printf("| c) |          | YEARS to DAYS   |\n");
    printf("+-----+\n");
    printf("| e) |          | YEARS to MONTHS |\n");
    printf("+-----+\n");

```

```

        printf("+-----+\n");
        printf("| Enter a choise:|      ");
        scanf(" %c", &choice);
        printf("+-----+\n");
        convertYEAR(value, choice);
        isValidUnit = 1;
    }
    else
    {
        printf("+-----+\n");
        printf("|      Invalid category.      |\n");
        printf("+-----+\n");
        isValidUnit = 1;
    }
}
printf("\033[0m");
return 0;
}

```

Makefile

```

CC = gcc
CFLAGS = -Wall -Wextra -std=c99
LDLAGS =

# Первая задача: сборка приложения для простого использования
app: coursework.o main.o
$(CC) $(LDLAGS) -o app coursework.o main.o

coursework.o: coursework.c coursework.h
$(CC) $(CFLAGS) -c coursework.c

main.o: main.c coursework.h
$(CC) $(CFLAGS) -c main.c

# Вторая задача: запуск тестов
test: coursework.o test.o
$(CC) $(LDLAGS) -o test coursework.o test.o

test.o: test.c coursework.h test.h
$(CC) $(CFLAGS) -c test.c

# Общие зависимости
coursework.o: coursework.h
main.o: coursework.h
test.o: coursework.h test.h

clean:
rm -f app test *.o

```

Test.c

```

#include <stdio.h>
#include "test.h"

```

```

#include <string.h>
#include <ctype.h>

int main()
{
    char units[10];
    printf("+-----+\n");
    printf("| Enter a unit:|");
    scanf(" %s", units);
    printf("+-----+\n");
    printf("\n\n");

    int P = 0;
    while (units[P] != '\0')
    {
        if (isdigit(units[P]))
        {
            printf("+-----+\n");
            printf("|input error: incorrect value entered|\n");
            printf("+-----+\n");
            return 1;
        }
        P++;
    }

    int isValidUnit = 0;
    while (!isValidUnit)
    {
        if (strcmp(units, "mm") == 0)
        {
            printf("| Choose conversion: | \n");
            printf("+-----+\n");
            printf("| a) | MM to CM | \n");
            printf("+-----+\n");
            printf("| b) | MM to M | \n");
            printf("+-----+\n");
            printf("| c) | MM to KM | \n");
            printf("+-----+\n");
            printf("| d) | MM to PD | \n");
            printf("+-----+\n");

            printf("+-----+\n");
            printf("| Enter a choose:|");
            scanf(" %c", &CTEST.choice);
            printf("+-----+\n");
            test_convertMM(CTEST.choice);
            isValidUnit = 1;
        }

        else if (strcmp(units, "cm") == 0)
        {
            printf("| Choose conversion: | \n");
            printf("+-----+\n");

```

```

printf("| Option |      | Conversion |\n");
printf("+-----+-----+\n");
printf("| a) |          | CM to M |\n");
printf("+-----+-----+\n");
printf("| b) |          | CM to MM |\n");
printf("+-----+-----+\n");
printf("| c) |          | CM to KM |\n");
printf("+-----+-----+\n");
printf("| d) |          | CM to DC |\n");
printf("+-----+-----+\n");

printf("\n\n");
printf("+-----+-----+\n");
printf("| Enter a choose:|      |");
scanf(" %c", &CTEST.choice);
printf("+-----+-----+\n");
test_convertSM(CTEST.choice);
printf("\n\n");
isValidUnit = 1;
}

else if (strcmp(units, "m") == 0)
{
printf("|      Choose conversion:      |\n");
printf("+-----+-----+\n");
printf("| Option |      | Conversion |\n");
printf("+-----+-----+\n");
printf("| a) |          | M to CM |\n");
printf("+-----+-----+\n");
printf("| b) |          | M to KM |\n");
printf("+-----+-----+\n");
printf("| c) |          | M to Ml |\n");
printf("+-----+-----+\n");
printf("| d) |          | M to Ft |\n");
printf("+-----+-----+\n");
printf("\n\n");
printf("+-----+-----+\n");
printf("| Enter a choose:|      |");
scanf(" %c", &CTEST.choice);
printf("+-----+-----+\n");
test_convertM(CTEST.choice);
printf("\n\n");
isValidUnit = 1;
}

else if (strcmp(units, "c") == 0)
{
printf("|      Choose conversion:      |\n");
printf("+-----+-----+\n");
printf("| a) |          | C to F |\n");
printf("+-----+-----+\n");
printf("| b) |          | C to K |\n");
printf("+-----+-----+\n");

printf("+-----+-----+\n");
printf("| Enter a choose:|      |");

```



```

scanf(" %c", &CTEST.choice);
printf("-----+\n");
test_convertC(CTEST.choice);
isValidUnit = 1;
}

else if (strcmp(units, "k") == 0)
{
    printf("| Choose conversion: |\n");
    printf("-----+\n");
    printf("| a) | | K to F |\n");
    printf("-----+\n");
    printf("| b) | | K to C |\n");
    printf("-----+\n");

    printf("-----+\n");
    printf("| Enter a choice:| ");
    scanf(" %c", &CTEST.choice);
    printf("-----+\n");
    test_convertK(CTEST.choice);
    isValidUnit = 1;
}

else if (strcmp(units, "f") == 0)
{
    printf("| Choose conversion: |\n");
    printf("-----+\n");
    printf("| a) | | F to K |\n");
    printf("-----+\n");
    printf("| b) | | F to C |\n");
    printf("-----+\n");

    printf("-----+\n");
    printf("| Enter a choice:| ");
    scanf(" %c", &CTEST.choice);
    printf("-----+\n");
    test_convertF(CTEST.choice);
    isValidUnit = 1;
}

else if (strcmp(units, "gr") == 0)
{
    printf("| Choose conversion: |\n");
    printf("-----+\n");
    printf("| a) | | GR to KG |\n");
    printf("-----+\n");
    printf("| b) | | GR to MG |\n");
    printf("-----+\n");
    printf("| b) | | GR to PD |\n");
    printf("-----+\n");
    printf("| d) | | GR to CR |\n");
    printf("-----+\n");
    printf("| e) | | GR to TN |\n");
    printf("-----+\n");
}

```

```

printf("+-----+\n");
printf("| Enter a choice:|      ");
scanf(" %c", &CTEST.choice);
printf("+-----+\n");
test_convertGR(CTEST.choice);
isValidUnit = 1;
}

else if (strcmp(units, "kg") == 0)
{
printf("| Choose conversion: | \n");
printf("+-----+\n");
printf("| a) |           | KG to GR | \n");
printf("+-----+\n");
printf("| b) |           | KG to PD | \n");
printf("+-----+\n");
printf("| c) |           | KG to MG | \n");
printf("+-----+\n");
printf("| d) |           | KG to CR | \n");
printf("+-----+\n");

printf("+-----+\n");
printf("| Enter a choice:|      ");
scanf(" %c", &CTEST.choice);
printf("+-----+\n");
test_convertKG(CTEST.choice);
isValidUnit = 1;
}

else if (strcmp(units, "hw") == 0)
{
printf("| Choose conversion: | \n");
printf("+-----+\n");
printf("| a) |           | HW to GR | \n");
printf("+-----+\n");
printf("| b) |           | HW to KG | \n");
printf("+-----+\n");
printf("| c) |           | HW to TN | \n");
printf("+-----+\n");
printf("| d) |           | HW to PD | \n");
printf("+-----+\n");

printf("+-----+\n");
printf("| Enter a choice:|      ");
scanf(" %c", &CTEST.choice);
printf("+-----+\n");
test_convertHW(CTEST.choice);
isValidUnit = 1;
}

else if (strcmp(units, "sec") == 0)
{
printf("| Choose conversion: | \n");
printf("+-----+\n");
printf("| a) |           | SEC to MIN | \n");

```

```

printf("+-----+\n");
printf("| b) |          | SEC to HOUR  |\n");
printf("+-----+\n");
printf("| c) |          | SEC to DAYS   |\n");
printf("+-----+\n");
printf("| d) |          | SEC to MONTHS |\n");
printf("+-----+\n");
printf("| e) |          | SEC to YEAR   |\n");
printf("+-----+\n");

printf("+-----+\n");
printf("| Enter a choose:|          |");
scanf(" %c", &CTEST.choice);
printf("+-----+\n");
test_convertSEC(CTEST.choice);
isValidUnit = 1;
}

else if (strcmp(units, "min") == 0)
{
    printf("| Choose conversion: |\n");
    printf("+-----+\n");
    printf("| a) |          | MIN to SEC   |\n");
    printf("+-----+\n");
    printf("| b) |          | MIN to HOURS |\n");
    printf("+-----+\n");
    printf("| c) |          | MIN to DAYS   |\n");
    printf("+-----+\n");
    printf("| d) |          | MIN to MONTHS |\n");
    printf("+-----+\n");
    printf("| e) |          | MIN to YEARS  |\n");
    printf("+-----+\n");

    printf("+-----+\n");
    printf("| Enter a choose:|          |");
    scanf(" %c", &CTEST.choice);
    printf("+-----+\n");
    test_convertMIN(CTEST.choice);
    isValidUnit = 1;
}

else if (strcmp(units, "hour") == 0)
{
    printf("| Choose conversion: |\n");
    printf("+-----+\n");
    printf("| a) |          | HOURS to SEC  |\n");
    printf("+-----+\n");
    printf("| b) |          | HOURS to MIN   |\n");
    printf("+-----+\n");
    printf("| c) |          | HOURS to DAYS  |\n");
    printf("+-----+\n");
    printf("| d) |          | HOURS to MONTHS |\n");
    printf("+-----+\n");
    printf("| e) |          | HOURS to YEARS  |\n");
    printf("+-----+\n");
}

```

```

    printf("+-----+\n");
    printf("| Enter a choice:|      ");
    scanf(" %c", &CTEST.choice);
    printf("+-----+\n");
    test_convertHOUR(CTEST.choice);
    isValidUnit = 1;
}

else if (strcmp(units, "day") == 0)
{
    printf("| Choose conversion: | \n");
    printf("+-----+\n");
    printf("| a) |           | DAYS to SEC | \n");
    printf("+-----+\n");
    printf("| b) |           | DAYS to MIN | \n");
    printf("+-----+\n");
    printf("| c) |           | DAYS to HOURS | \n");
    printf("+-----+\n");
    printf("| d) |           | DAYS to MONTHS | \n");
    printf("+-----+\n");
    printf("| e) |           | DAYS to YEARS | \n");
    printf("+-----+\n");

    printf("+-----+\n");
    printf("| Enter a choice:|      ");
    scanf(" %c", &CTEST.choice);
    printf("+-----+\n");
    test_convertDAY(CTEST.choice);
    isValidUnit = 1;
}

else if (strcmp(units, "month") == 0)
{
    printf("| Choose conversion: | \n");
    printf("+-----+\n");
    printf("| a) |           | MONTHS to SEC | \n");
    printf("+-----+\n");
    printf("| b) |           | MONTHS to MIN | \n");
    printf("+-----+\n");
    printf("| c) |           | MONTHS to HOURS | \n");
    printf("+-----+\n");
    printf("| c) |           | MONTHS to DAYS | \n");
    printf("+-----+\n");
    printf("| e) |           | MONTHS to YEARS | \n");
    printf("+-----+\n");

    printf("+-----+\n");
    printf("| Enter a choice:|      ");
    scanf(" %c", &CTEST.choice);
    printf("+-----+\n");
    test_convertMONTH(CTEST.choice);
    isValidUnit = 1;
}

```

```

else if (strcmp(units, "year") == 0)
{
    printf("|    Choose conversion:    |\n");
    printf("+-----+\n");
    printf("| a) |          | YEARS to SEC |\n");
    printf("+-----+\n");
    printf("| b) |          | YEARS to MIN |\n");
    printf("+-----+\n");
    printf("| c) |          | YEARS to HOURS |\n");
    printf("+-----+\n");
    printf("| c) |          | YEARS to DAYS |\n");
    printf("+-----+\n");
    printf("| e) |          | YEARS to MONTHS |\n");
    printf("+-----+\n");

    printf("+-----+\n");
    printf("| Enter a choice:|          |");
    scanf(" %c", &CTEST.choice);
    printf("+-----+\n");
    test_convertYEAR(CTEST.choice);
    isValidUnit = 1;
}
else
{
    printf("+-----+\n");
    printf("|          Invalid category.    |\n");
    printf("+-----+\n");
    isValidUnit = 1;
}
}

return 0;
}

```

test.c

```

#include <stdio.h>
#include <stdlib.h>
#include "../src/LibConvert/coursework.h"
#include <ctest.h>
#include "../thirdparty/ctest.h" // заголовочный файл с функциями

//проверка сантиметров
CTEST(CONVERT_SM, test_A) {
    float result = convertSM(100, 'a');
    ASSERT_DBL_NEAR(1.0, result);
}

CTEST(CONVERT_SM, test_B) {
    float result = convertSM(20, 'b');
    ASSERT_DBL_NEAR(200.0, result);
}

CTEST(CONVERT_SM, test_C) {

```

```
float result = convertSM(10000.00, 'c');  
ASSERT_DBL_NEAR_TOL(0.1, result, 1e-6);  
}
```

```
CTEST(CONVERT_SM, test_D) {  
    float result = convertSM(20, 'd');  
    ASSERT_DBL_NEAR(2, result);  
}
```

```
//проверка метров  
CTEST(CONVERTM, testA) {  
    float result = convertationM(100, 'a');  
    ASSERT_DBL_NEAR(10000, result);  
}
```

```
CTEST(CONVERTM, testB) {  
    float result = convertationM(1000, 'b');  
    ASSERT_DBL_NEAR(1, result);  
}
```

```
CTEST(CONVERTM, testC) {  
    float result = convertationM(4276, 'c');  
    ASSERT_DBL_NEAR(4, result);  
}
```

```
CTEST(CONVERTM, testD) {  
    float result = convertationM(4, 'd');  
    ASSERT_DBL_NEAR_TOL(13.12, result, 0.01);  
}
```

```
//проверка миллиметров  
CTEST(CONVERTMM, testA) {  
    float result = convertMM(100, 'a');  
    ASSERT_DBL_NEAR(10, result);  
}
```

```
CTEST(CONVERTMM, testB) {  
    float result = convertMM(1000, 'b');  
    ASSERT_DBL_NEAR(1, result);  
}
```

```
CTEST(CONVERTMM, testC) {  
    float result = convertMM(10000, 'c');  
    ASSERT_DBL_NEAR_TOL(10000000000.00, result, 0.01);  
}
```

```
CTEST(CONVERTMM, testD) {  
    float result = convertMM(1000, 'd');  
    ASSERT_DBL_NEAR_TOL(3.28084, result, 0.01);  
}
```

```
//проверка цельсия
CTEST(CONVERTC, testA) {
    float result = convertC(6, 'a');
    ASSERT_DBL_NEAR_TOL(42.8, result, 0.01);
}
```

```
CTEST(CONVERTC, testB) {
    float result = convertC(100, 'b');
    ASSERT_DBL_NEAR(373, result);
}
```

```
//проверка кельвины
CTEST(CONVERTK, testA) {
    float result = convertK(300, 'a');
    ASSERT_DBL_NEAR_TOL(80.6, result, 0.01);
}
```

```
CTEST(CONVERTK, testB) {
    float result = convertK(100, 'b');
    ASSERT_DBL_NEAR(-173, result);
}
```

```
//проверка фаренгейт
CTEST(CONVERTF, testA) {
    float result = convertF(300, 'a');
    ASSERT_DBL_NEAR_TOL(422.04, result, 0.01);
}
```

```
CTEST(CONVERTF, testB) {
    float result = convertF(32, 'b');
    ASSERT_DBL_NEAR(0, result);
}
```

```
//проверка килограммов
CTEST(CONVERTKG, testA) {
    float result = convertKG(1, 'a');
    ASSERT_DBL_NEAR(1000, result);
}
```

```
CTEST(CONVERTKG, testB) {
    float result = convertKG(4, 'b');
    ASSERT_DBL_NEAR_TOL(8.82, result, 0.01);
}
```

```
CTEST(CONVERTKG, testC) {
    float result = convertKG(1, 'c');
    ASSERT_DBL_NEAR_TOL(1000000.0, result, 0.01);
}
```

```
CTEST(CONVERTKG, testD) {  
    float result = convertKG(1, 'd');  
    ASSERT_DBL_NEAR(5000, result);  
}
```

```
//проверка граммов  
CTEST(CONVERTGR, testA) {  
    float result = convertGR(10000, 'a');  
    ASSERT_DBL_NEAR(10, result);  
}
```

```
CTEST(CONVERTGR, testB) {  
    float result = convertGR(1, 'b');  
    ASSERT_DBL_NEAR(1000, result);  
}
```

```
CTEST(CONVERTGR, testC) {  
    float result = convertGR(400, 'c');  
    ASSERT_DBL_NEAR_TOL(0.88, result, 0.01);  
}
```

```
CTEST(CONVERTGR, testD) {  
    float result = convertGR(10, 'd');  
    ASSERT_DBL_NEAR(50, result);  
}
```

```
CTEST(CONVERTGR, testE) {  
    float result = convertGR(15, 'e');  
    ASSERT_DBL_NEAR_TOL(0.0015, result, 0.01);  
}
```

```
//проверка центнеров  
CTEST(CONVERTHW, testA) {  
    float result = convertHW(1, 'a');  
    ASSERT_DBL_NEAR(100000, result);  
}
```

```
CTEST(CONVERTHW, testB) {  
    float result = convertHW(1, 'b');  
    ASSERT_DBL_NEAR(100, result);  
}
```

```
CTEST(CONVERTHW, testC) {  
    float result = convertHW(100, 'c');  
    ASSERT_DBL_NEAR(10, result);  
}
```

```
//проверка секунд  
CTEST(CONVERTSEC, testA) {  
    float result = convertSEC(360, 'a');
```



```
    ASSERT_DBL_NEAR(6, result);  
}
```

```
CTEST(CONVERTSEC, testB) {  
    float result = convertSEC(3600, 'b');  
    ASSERT_DBL_NEAR(1, result);  
}
```

```
//проверка минут  
CTEST(CONVERTMIN, testA) {  
    float result = convertMIN(6, 'a');  
    ASSERT_DBL_NEAR(360, result);  
}
```

```
CTEST(CONVERTMIN, testB) {  
    float result = convertMIN(60, 'b');  
    ASSERT_DBL_NEAR(1, result);  
}
```

```
//проверка часов  
CTEST(CONVERTHOUR, testA) {  
    float result = convertHOUR(1, 'a');  
    ASSERT_DBL_NEAR(3600, result);  
}
```

```
CTEST(CONVERTHOUR, testB) {  
    float result = convertHOUR(1, 'b');  
    ASSERT_DBL_NEAR(60, result);  
}
```

```
//проверка дней  
CTEST(CONVERTDAY, testA) {  
    float result = convertDAY(1, 'a');  
    ASSERT_DBL_NEAR(86400, result);  
}
```

```
CTEST(CONVERTDAY, testB) {  
    float result = convertDAY(10, 'b');  
    ASSERT_DBL_NEAR(14400, result);  
}
```

```
CTEST(CONVERT_DAY, test_C) {  
    float result = convertDAY(24, 'c');  
    ASSERT_DBL_NEAR(576, result);  
}
```

```
//проверка месяца  
CTEST(CONVERT_MONTH, test_A) {  
    float result = convertMONTH(1, 'a');  
    ASSERT_DBL_NEAR_TOL(2628000.0000000, result, 0.1);  
}
```

```
}
```

```
CTEST(CONVERT_MONTH, test_B) {  
    float result = convertMONTH(1, 'b');  
    ASSERT_DBL_NEAR(43800, result);  
}
```

```
CTEST(CONVERT_MONTH, test_C) {  
    float result = convertMONTH(1, 'c');  
    ASSERT_DBL_NEAR(730, result);  
}
```

```
CTEST(CONVERT_MONTH, test_D) {  
    float result = convertMONTH(15, 'd');  
    ASSERT_DBL_NEAR_TOL(456.26, result, 0.01);  
}
```

```
CTEST(CONVERT_MONTH, test_E) {  
    float result = convertMONTH(12, 'e');  
    ASSERT_DBL_NEAR(1, result);  
}  
//для года  
CTEST(CONVERT_YEAR, test_A) {  
    float result = convertYEAR(15, 'a');  
    ASSERT_DBL_NEAR_TOL(473100000.00, result, 0.01);  
}
```

```
CTEST(CONVERT_YEAR, test_B) {  
    float result = convertYEAR(10, 'b');  
    ASSERT_DBL_NEAR_TOL(5256000.00, result, 0.01);  
}
```

```
CTEST(CONVERT_YEAR, test_C) {  
    float result = convertYEAR(1, 'c');  
    ASSERT_DBL_NEAR(8760, result);  
}
```

```
CTEST(CONVERT_YEAR, test_D) {  
    float result = convertYEAR(1, 'd');  
    ASSERT_DBL_NEAR(365, result);  
}
```

```
CTEST(CONVERT_YEAR, test_E) {  
    float result = convertYEAR(1, 'e');  
    ASSERT_DBL_NEAR(12, result);  
}
```

mainTest.c

```
#define CTEST_MAIN
#include <ctest.h>
// uncomment lines below to enable/disable features. See README.md for details
#define CTEST_SEGFAULT
//#define CTEST_NO_COLORS
//#define CTEST_COLOR_OK

int main(int argc, const char *argv[])
{
    int result = ctest_main(argc, argv);
    return result;
}
```

Makefile

```
CC = gcc
```

```
APP_NAME = Convert
```

```
LIB_NAME = LibConvert
```

```
TEST_NAME = test
```

```
CFLAGS = -Wall -Wextra -Werror
```

```
CPPFLAGS = -I src -MP -MMD
```

```
CPPFLAGST = -I thirdparty -MP -MMD
```

```
LDFLAGS =
```

```
LDLIBS =
```

```
BIN_DIR = bin
```

```
OBJ_DIR = obj
```

```
SRC_DIR = src
```

```
TEST_DIR = test
```

```
APP_PATH = $(BIN_DIR)/$(APP_NAME)
```

```
TEST_PATH = $(BIN_DIR)/$(TEST_NAME)
```

```
LIB_PATH = $(OBJ_DIR)/$(SRC_DIR)/$(LIB_NAME)/$(LIB_NAME).a
```

```
SRC_EXT = c
```

```
APP_SOURCES = $(shell find $(SRC_DIR)/$(APP_NAME) -name '*.$(SRC_EXT)')
```

```
APP_OBJECTS = $(APP_SOURCES:$(SRC_DIR)/%.${SRC_EXT}=$(OBJ_DIR)/$(SRC_DIR)/%.o)
```

```
TEST_SOURCES = $(shell find $(TEST_DIR) -name '*.$(SRC_EXT)')
```

```
TEST_OBJECTS = $(TEST_SOURCES:$(TEST_DIR)/%.${SRC_EXT}=$(OBJ_DIR)/$(TEST_DIR)/%.o)
```

```
LIB_SOURCES = $(shell find $(SRC_DIR)/$(LIB_NAME) -name '*.$(SRC_EXT)')
LIB_OBJECTS = $(LIB_SOURCES:$(SRC_DIR)/%.$(SRC_EXT)=$(OBJ_DIR)/$(SRC_DIR)/%.o)
```

```
DEPS = $(APP_OBJECTS:.o=.d) $(LIB_OBJECTS:.o=.d) $(TEST_OBJECTS:.o=.d)
```

```
.PHONY: all
all: $(APP_PATH)
```

```
-include $(DEPS)
```

```
$(APP_PATH): $(APP_OBJECTS) $(LIB_PATH)
    $(CC) $(CFLAGS) $(CPPFLAGS) $^ -o $@ -lm $(LDFLAGS) $(LDLIBS)
```

```
$(LIB_PATH): $(LIB_OBJECTS)
    ar rcs $@ $^
```

```
$(OBJ_DIR)/%.o: %.c
    $(CC) -c $(CFLAGS) $(CPPFLAGS) $(CPPFLAGST) $< -o $@
```

```
.PHONY: run clean test run_test
```

```
run:
    ./bin/Convert
```

```
run_test:
    ./bin/test
```

```
clean:
    $(RM) $(APP_PATH) $(OBJ_DIR)/**/*.[aod]
    $(RM) bin/*.exe
    $(RM) $(OBJ_DIR)/$(TEST_DIR)/**/*.[aod]
```

```
test: $(TEST_PATH)
```

```
-include $(DEPS)
```

```
$(TEST_PATH): $(TEST_OBJECTS) $(LIB_PATH)
    $(CC) $(CFLAGS) $(CPPFLAGST) $^ -o $@ -lm $(LDFLAGS) $(LDLIBS)
```