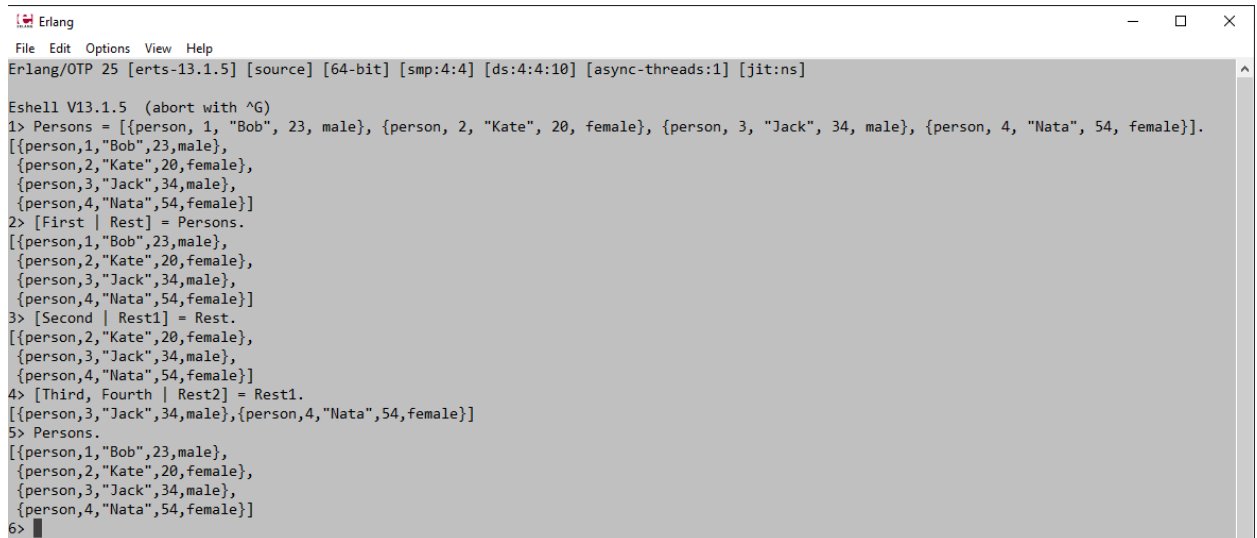


Выполнил: Белкин Никита

Задание 1.



```
Erlang
File Edit Options View Help
Erlang/OTP 25 [erts-13.1.5] [source] [64-bit] [smp:4:4] [ds:4:4:10] [async-threads:1] [jit:ns]

Eshell V13.1.5 (abort with ^G)
1> Persons = [{person, 1, "Bob", 23, male}, {person, 2, "Kate", 20, female}, {person, 3, "Jack", 34, male}, {person, 4, "Nata", 54, female}].
[{person,1,"Bob",23,male},
 {person,2,"Kate",20,female},
 {person,3,"Jack",34,male},
 {person,4,"Nata",54,female}]
2> [First | Rest] = Persons.
[{person,1,"Bob",23,male},
 {person,2,"Kate",20,female},
 {person,3,"Jack",34,male},
 {person,4,"Nata",54,female}]
3> [Second | Rest1] = Rest.
[{person,2,"Kate",20,female},
 {person,3,"Jack",34,male},
 {person,4,"Nata",54,female}]
4> [Third, Fourth | Rest2] = Rest1.
[{person,3,"Jack",34,male},{person,4,"Nata",54,female}]
5> Persons.
[{person,1,"Bob",23,male},
 {person,2,"Kate",20,female},
 {person,3,"Jack",34,male},
 {person,4,"Nata",54,female}]
6>
```

В этом задании мы сначала создаем список кортежей Persons, содержащий информацию о 4 людях. Затем выполняем команды:

1. [First | Rest] = Persons.

Эта команда присвоит первый элемент Persons переменной First, а остальную часть списка - Rest. В данном случае First - это кортеж, содержащий информацию о первом человеке в списке ({person, 1, "Bob", 23, male}), а Rest - это список остальных кортежей в Persons.

2. [Second | Rest1] = Rest.

Эта команда присвоит первый элемент Rest переменной Second, а остальной список - Rest1. В данном случае Second - это кортеж, содержащий информацию о втором человеке в списке ({person, 2, "Kate", 20, female}), а Rest1 - это список оставшихся кортежей в Rest.

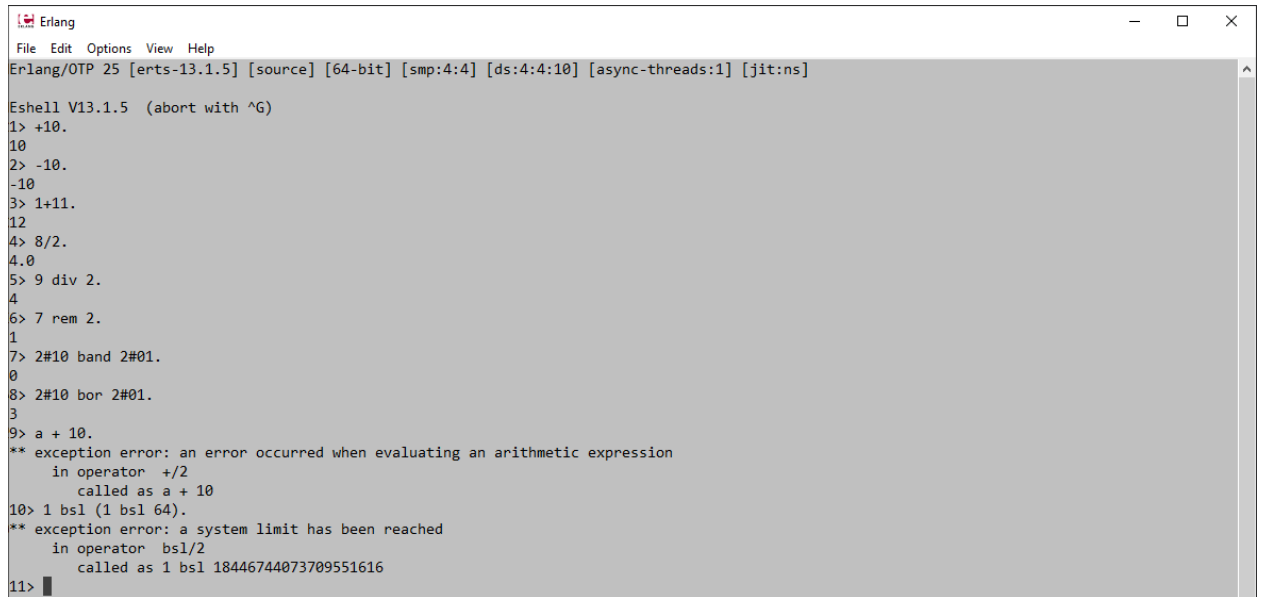
3. [Third, Fourth | Rest2] = Rest1.

Эта команда присвоит первые два элемента Rest1 переменным Third и Fourth, а остальной список - Rest2. В данном случае Third - это кортеж, содержащий информацию о третьем человеке в списке ({person, 3, "Jack", 34, male}), Fourth - кортеж, содержащий информацию о четвертом человеке в списке ({person, 4, "Nata", 54, female}), а Rest2 - пустой список.

4. Persons.

Проверили, что список Persons не изменился.

Задание 2.



```
Erlang
File Edit Options View Help
Erlang/OTP 25 [erts-13.1.5] [source] [64-bit] [smp:4:4] [ds:4:4:10] [async-threads:1] [jit:ns]

Eshell V13.1.5 (abort with ^G)
1> +10.
10
2> -10.
-10
3> 1+11.
12
4> 8/2.
4.0
5> 9 div 2.
4
6> 7 rem 2.
1
7> 2#10 band 2#01.
0
8> 2#10 bor 2#01.
3
9> a + 10.
** exception error: an error occurred when evaluating an arithmetic expression
   in operator  +/2
   called as a + 10
10> 1 bsl (1 bsl 64).
** exception error: a system limit has been reached
   in operator  bsl/2
   called as 1 bsl 18446744073709551616
11>
```

- +10.

Это выражение прибавляет 10 к неявному начальному значению (0) и возвращает результат.

- -10.

Это выражение вычитает 10 из неявного начального значения (0) и возвращает результат.

- 1+11.

Это выражение складывает 1 и 11 вместе и возвращает результат.

- 8/2.

Это выражение делит 8 на 2 и возвращает результат. Поскольку оба операнда являются целыми числами, то результатом является число с типом float.

- 9 div 2.

Это выражение делит 9 на 2 с помощью целочисленного деления и возвращает результат.

- 7 rem 2.

Это выражение вычисляет целочисленный остаток от деления 7 на 2 и возвращает результат.

- 2#10 band 2#01.

Это выражение выполняет операцию побитового И между двоичными числами 2#10 (это 2 в десятичной системе) и 2#01 (это 1 в десятичной системе) и возвращает результат. В данном случае результат равен 0, так как в двоичном представлении нет ни одной 1 в той же позиции.

- `2#10 bor 2#01`.

Это выражение выполняет операцию побитового ИЛИ между двоичными числами `2#10` (это 2 в десятичной системе) и `2#01` (это 1 в десятичной системе) и возвращает результат. В данном случае двоичное представление результата равно `2#11`, что в десятичной системе равно 3.

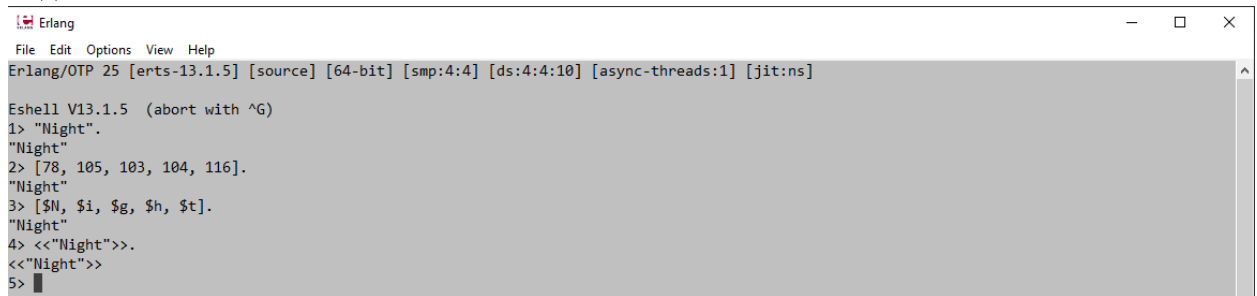
- `a + 10`.

Это выражение пытается добавить атом `a` к числу 10, что не является допустимой арифметической операцией. Erlang выдает исключение с сообщением об ошибке, поскольку выражение недействительно.

- `1 bsl (1 bsl 64)`.

Это выражение пытается сдвинуть двоичное представление числа 1 влево на 2^{64} бита, что находится вне диапазона допустимых значений сдвига. Erlang выдает исключение с сообщением об ошибке, поскольку выражение недействительно.

Задание 3.



```
Erlang
File Edit Options View Help
Erlang/OTP 25 [erts-13.1.5] [source] [64-bit] [smp:4:4] [ds:4:4:10] [async-threads:1] [jit:ns]

Eshell V13.1.5 (abort with ^G)
1> "Night".
"Night"
2> [78, 105, 103, 104, 116].
[78, 105, 103, 104, 116]
3> [$N, $i, $g, $h, $t].
[$N, $i, $g, $h, $t]
4> <<"Night">>.
<<"Night">>
5> <<[78, 105, 103, 104, 116]>>.
<<[78, 105, 103, 104, 116]>>
```

Для `binary` можно еще написать такое выражение:

```
5> <<78, 105, 103, 104, 116>>.
<<"Night">>
```