

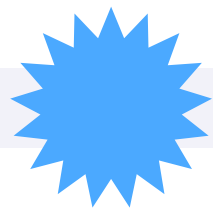
Программирование на C++



Минцифры
России

UCHi **DOMA**

20.35
УНИВЕРСИТЕТ

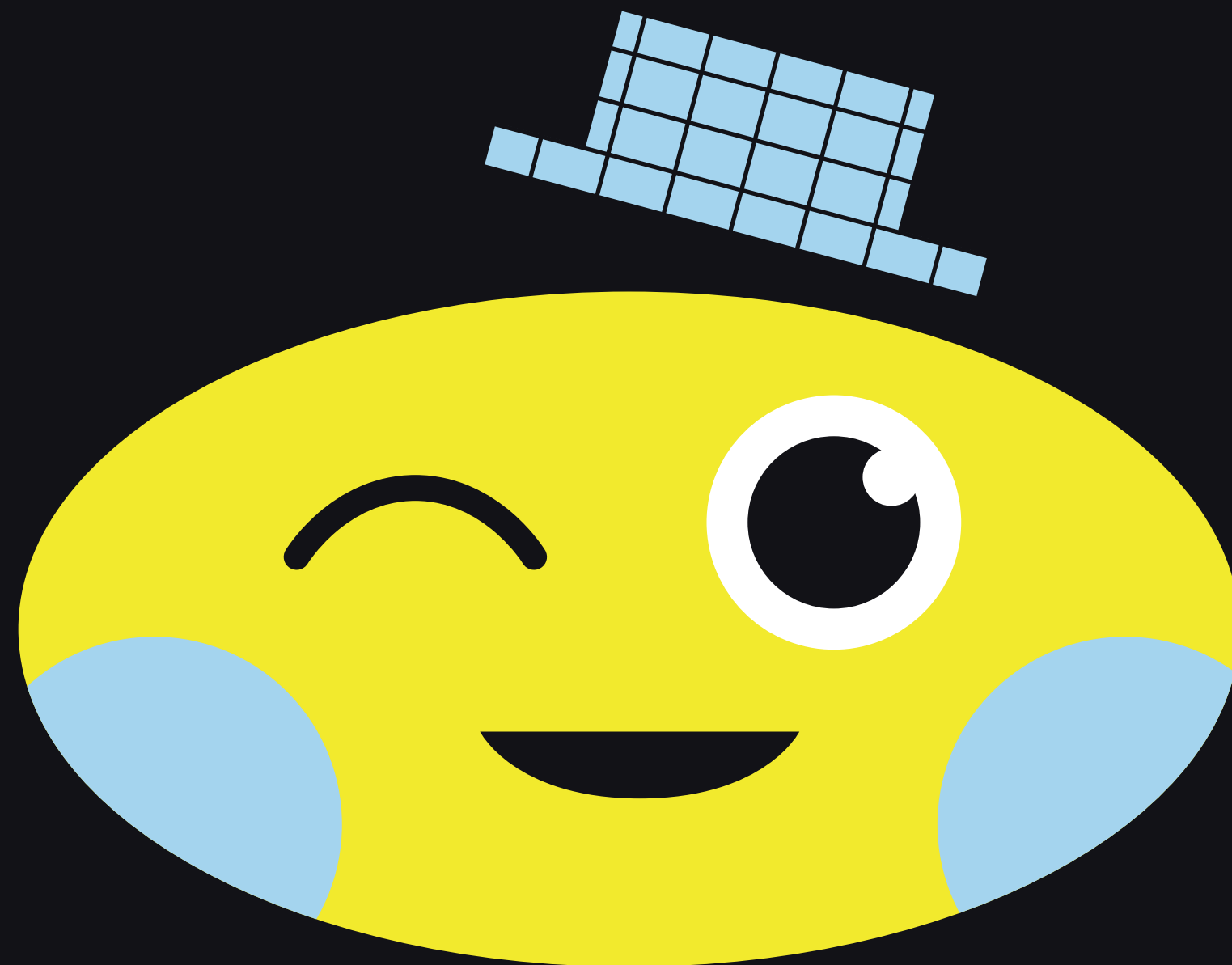


Модуль 2. Урок 3

Рекурсия



Привет!



проверка готовности



Видим и слышим друг друга без помех



Не опаздываем и не отвлекаемся



Сидим прямо



Улыбаемся, если всё ок

Как домашка?



Какие были трудности?



Какие остались вопросы?



Сколько заданий выполнено?



Разомнёмся



Что будет выведено на экран в результате работы программы если ввести 4, а затем 5?

```
1  #include <stdio.h>
2  int x=0, y=0;
3  int sum(int x, int y)
4  {
5      return (x+y);
6  }
7  int main()
8  {
9      printf("x= ");
10     scanf("%d", &x);
11     printf("y= ");
12     scanf("%d", &y);
13     printf("x+y= %d", sum(2,3));
14     return 0;
15 }
```

Разомнёмся



Что будет выведено на экран в результате работы программы если ввести 4, а затем 5?

```
1  #include <stdio.h>
2  int x=0, y=0;
3  int sum(int x, int y)
4  {
5      return (x+y);
6  }
7  int main()
8  {
9      printf("x= ");
10     scanf("%d", &x);
11     printf("y= ");
12     scanf("%d", &y);
13     printf("x+y= %d", sum(2,3));
14     return 0;
15 }
```

Результат работы программы:

x=4

y=5

x+y=5

Разомнёмся



Что будет выведено на экран в результате работы программы если ввести 4, а затем 5?

```
1  #include <stdio.h>
2  int x=0, y=0;
3  int sum()
4  {
5      return (x+y);
6  }
7  int main()
8  {
9      printf("x= ");
10     scanf("%d", &x);
11     printf("y= ");
12     scanf("%d", &y);
13     printf("x+y= %d", sum());
14     return 0;
15 }
```


Разомнёмся



Что будет выведено на экран в результате работы программы если ввести 4, а затем 5?

```
1  #include <stdio.h>
2  int x=0, y=0;
3  int sum()
4  {
5      return (x+y);
6  }
7  int main()
8  {
9      printf("x= ");
10     scanf("%d", &x);
11     printf("y= ");
12     scanf("%d", &y);
13     printf("x+y= %d", sum());
14     return 0;
15 }
```

Результат работы программы:

x=4

y=5

x+y=9

Задача



Пользователь вводит значение x ,
выведите на экран цифры от x до 1?



Задача



Пользователь вводит значение x ,
выведите на экран цифры от x до 1?

```
1  #include <stdio.h>
2  int main()
3  {
4      int x;
5      printf("x= ");
6      scanf("%d", &x);
7      for (int i=x; i>=1; i--)
8          printf("%d\n", i);
9      return 0;
10 }
```

Результат работы программы:

x=5

5

4

3

2

1

Задача



Пользователь вводит значение x ,
выведите на экран цифры от x до 1?

```
1  #include <stdio.h>
2  int main()
3  {
4      int x;
5      printf("x= ");
6      scanf("%d", &x);
7      for (int i=x; i>=1; i--)
8          printf("%d\n", i);
9      return 0;
10 }
```

?

Как решить
без использования
цикла?



Цели урока



изучить рекурсию



отработать на практике
составление рекурсивных
алгоритмов на Си



Рекурсия

Рекурсивная функция — это функция, которая вызывает сама себя.



Рекурсия — вызов функции из самой функции

Пример



Программа выводит цифры от 0 до 3

```
1  #include <stdio.h>
2  void func(int num)
3  {
4      if (num > 0) func(num - 1);
5      printf("%d\n", num);
6  }
7  int main()
8  {
9      func(3);
10     return 0;
11 }
```

Результат работы программы:

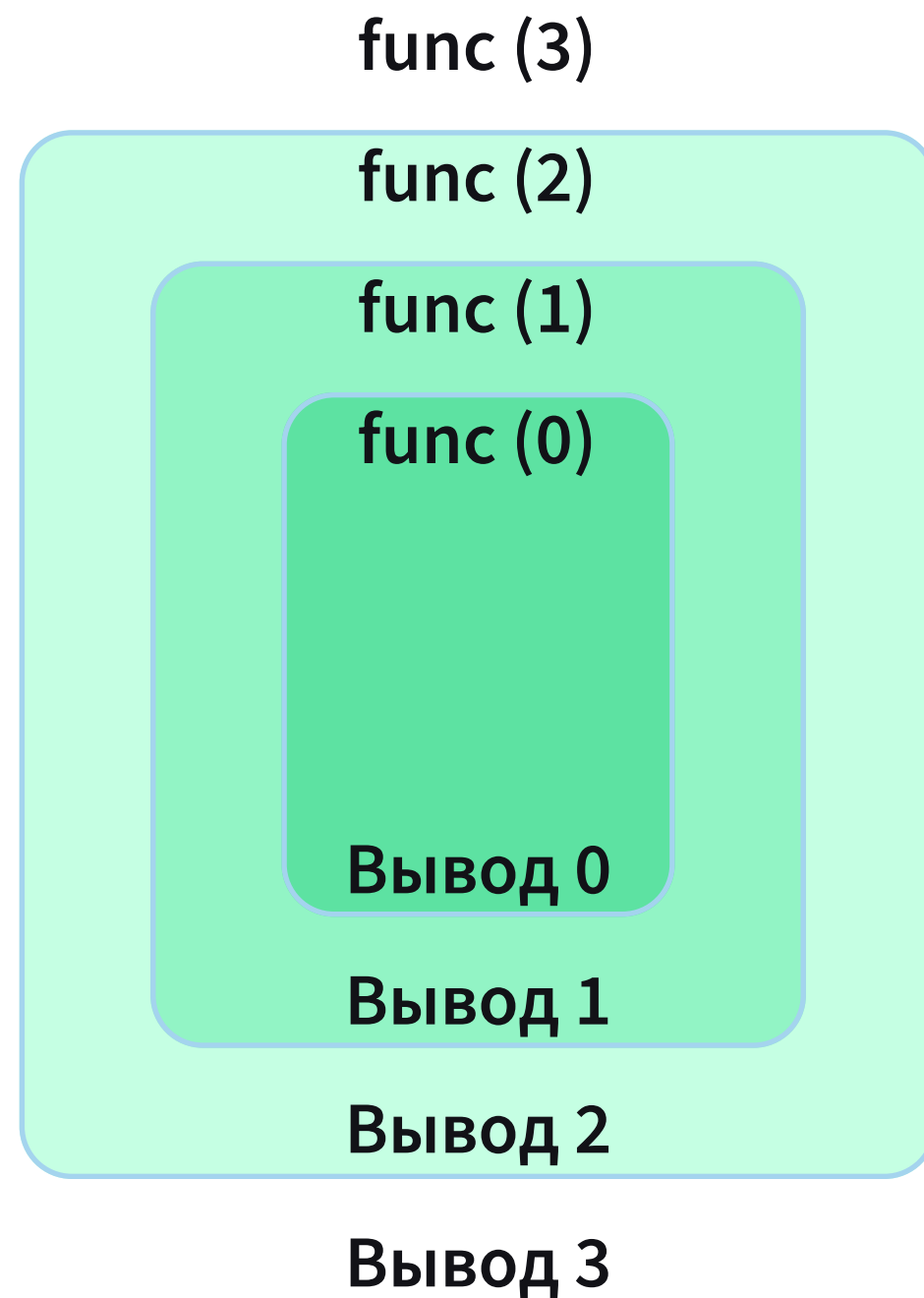
```
0
1
2
3
```

Пример



Программа выводит цифры от 0 до 3

```
1  #include <stdio.h>
2  void func(int num)
3  {
4      if (num > 0) func(num - 1);
5      printf("%d\n", num);
6  }
7  int main()
8  {
9      func(3);
10     return 0;
11 }
```



Задача



Пользователь вводит значение x, выведите на экран цифры от x до 1?

```
1  #include <stdio.h>
2  int countdown(int i)
3  {
4      printf("%d\n",i);
5      if (i<=1) return 0;
6      else countdown(i - 1);
7      return 0;
8  }
9  int main()
10 {
11     int x;
12     printf("x= ");
13     scanf("%d", &x);
14     countdown(x);
15     return 0;
16 }
```



Используем рекурсию



Пример рекурсии



Пользователь вводит натуральное число a , вычислить $a!$ (a факториал)

```
1  #include <stdio.h>
2  int fact(int num) // вычисление факториала числа num
3  {
4      if (num <= 1) return 1; // если число не больше 1, возвращаем 1
5      else return num*fact(num - 1); // рекурсивный вызов для числа на 1 меньше
6  }
7  // Главная функция
8  int main()
9  {
10     int a, r;
11     printf("a= ");
12     scanf("%d", &a);
13     r = fact(a); // вызов функции: num=a
14     printf("%d! = %d", a, r);
15     return 0;
16 }
```

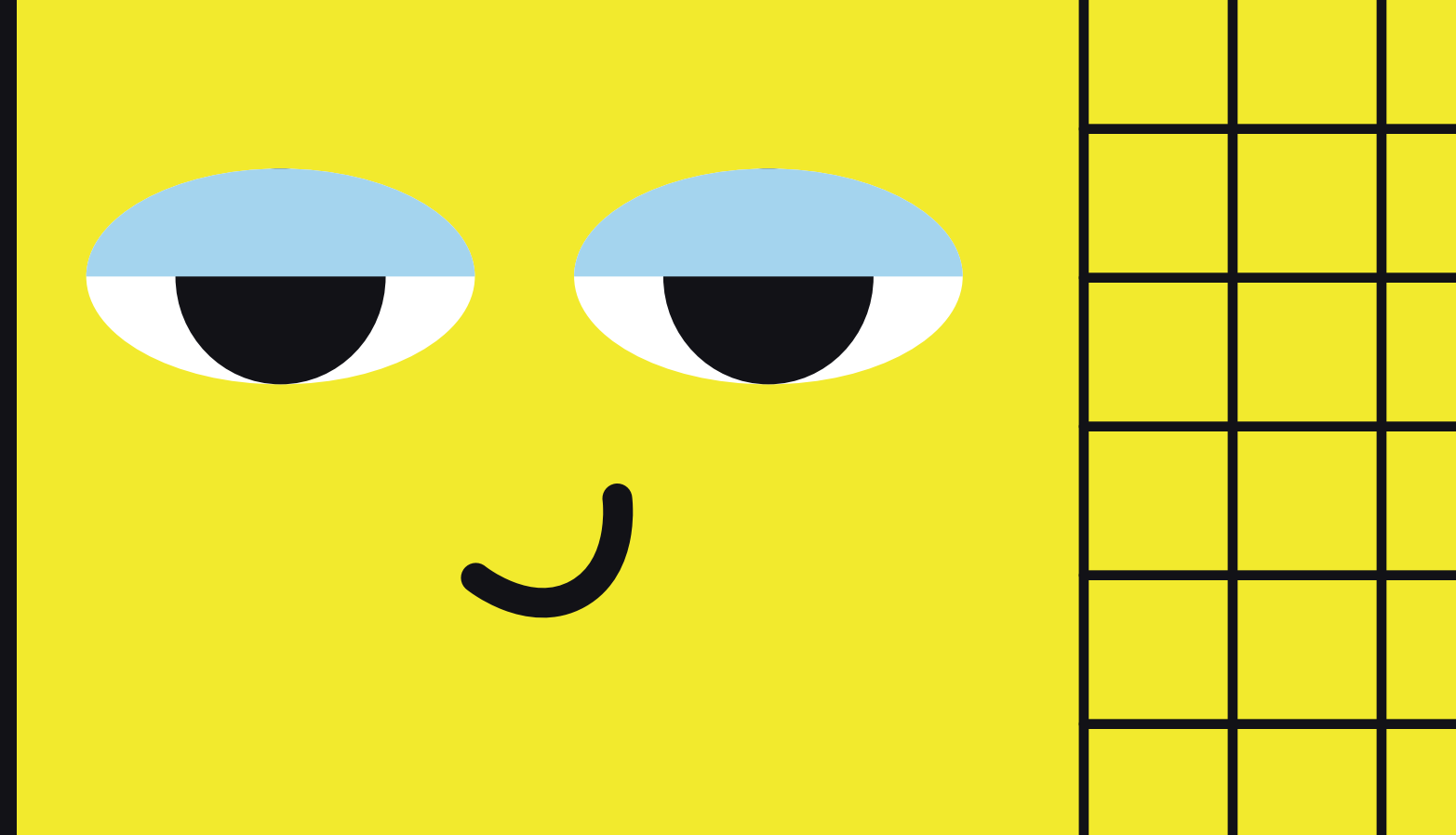
Пример рекурсии



Результат выполнения программы

$a = 3$

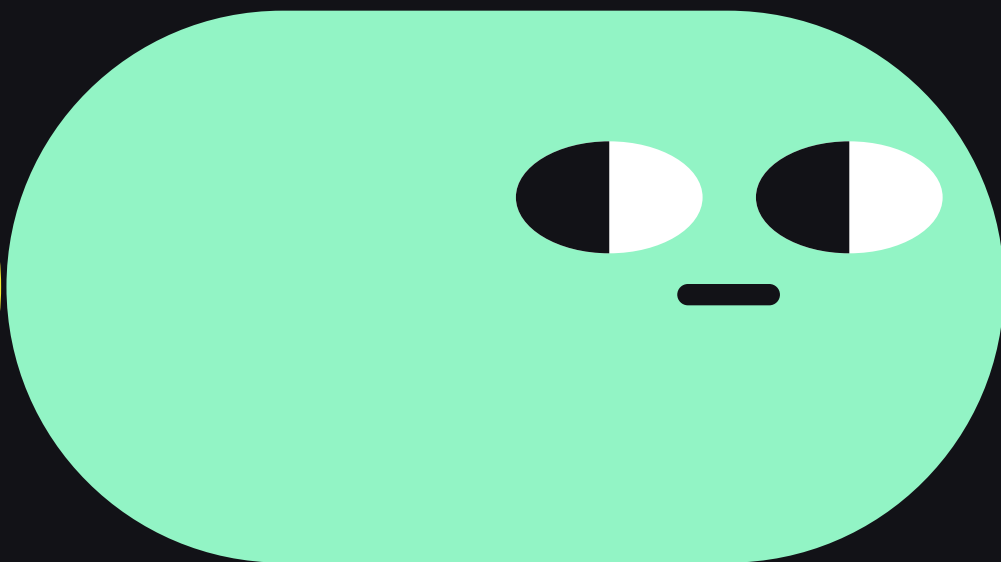
$3! = 6$



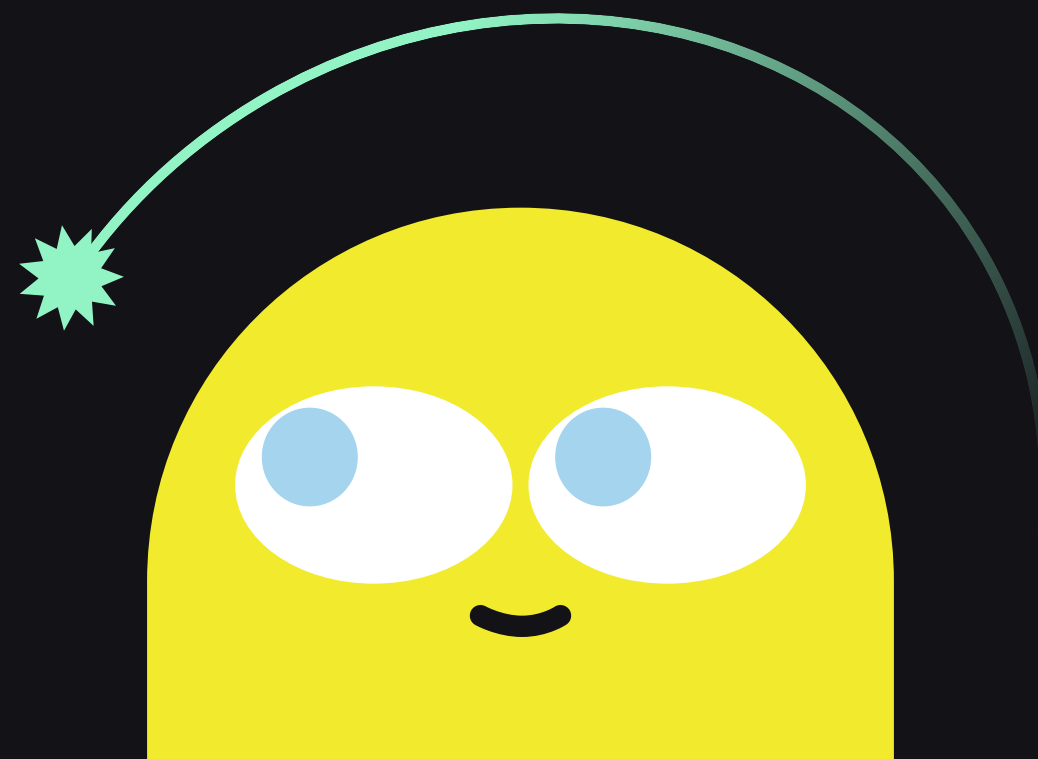
Практика

перерыв

физкультминутка



Смотрим вверх–вниз, вправо–влево



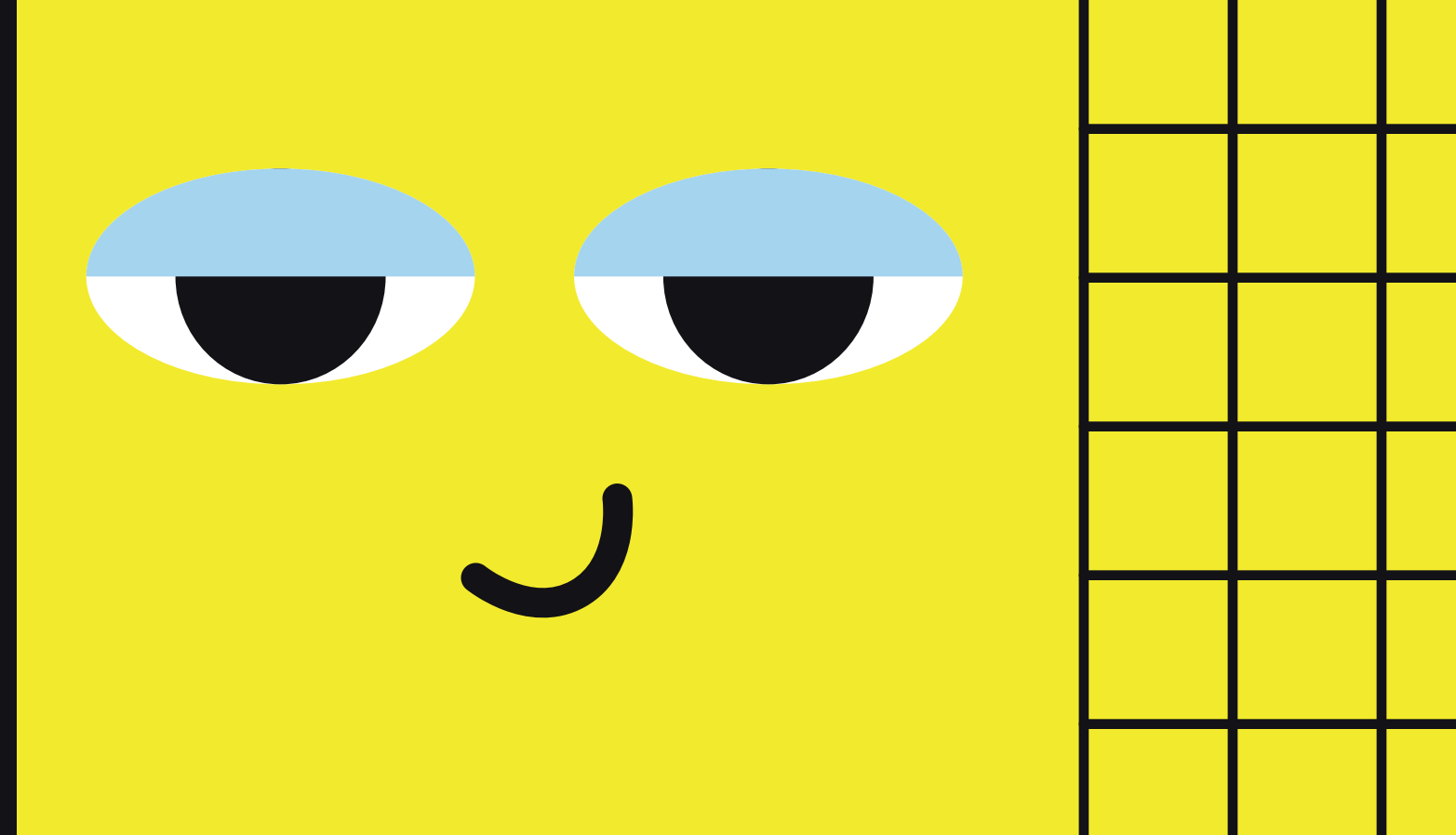
Вращаем по кругу туда–обратно



Крепко зажимаемся



Быстро моргаем



Практика

1

Закрепление

Алгоритм вычисления значения функции $F(n)$, где n — натуральное число, задан следующими соотношениями:

$$F(1) = 1$$

$$F(n) = F(n-1) * n, \text{ при } n > 1$$

Чему равно значение функции $F(5)$?

```
1  #include <stdio.h>
2  int F(int n)
3  {
4      if (n==1) return 1;
5      else return F(n-1)*n;
6  }
7  int main()
8  {
9      int n;
10     scanf("%d", &n);
11     printf("%d\n", F(n));
12     return 0;
13 }
```

1

Закрепление

Алгоритм вычисления значения функции $F(n)$, где n — натуральное число, задан следующими соотношениями:

$$F(1) = 1$$

$$F(n) = F(n-1) * n, \text{ при } n > 1$$

Чему равно значение функции $F(5)$?

Последовательно находим:

$$F(2) = F(1) * 2 = 2,$$

$$F(3) = F(2) * 3 = 6,$$

$$F(4) = F(3) * 4 = 24,$$

$$F(5) = F(4) * 5 = 120.$$



Использование функции позволяет вычислить факториал числа n — произведение натуральных чисел от 1 до n . Тем самым,
 $F(5) = 1 * 2 * 3 * 4 * 5 = 120$.



Подведём итоги



изучили рекурсию



отработали на практике
составление рекурсивных
алгоритмов на Си

Оцени сложность урока

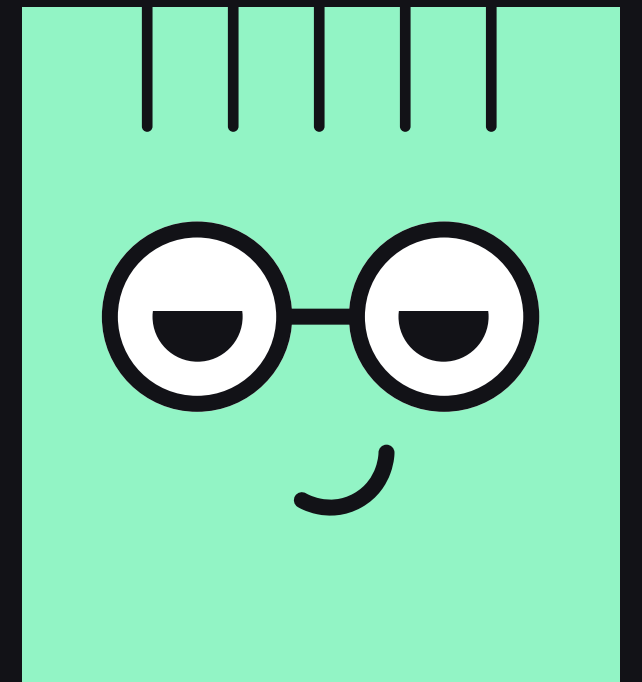
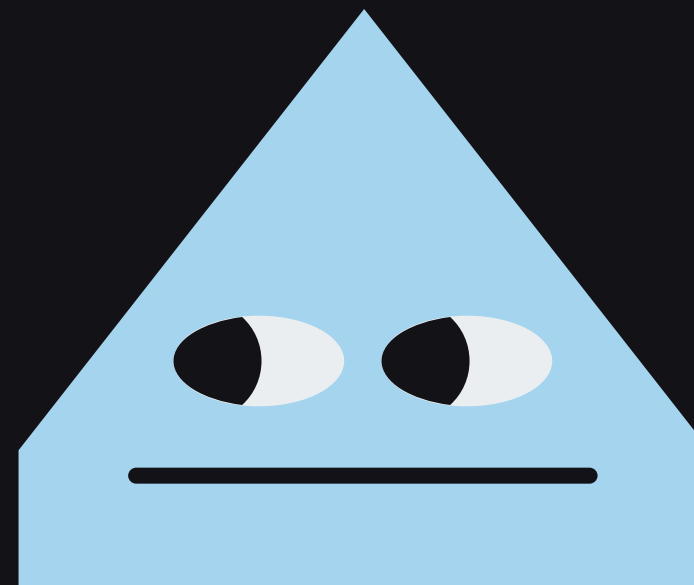
1 если тебе было совсем просто

2 было достаточно просто, но ты узнал(а) что-то новое

3 было не очень просто, но достаточно комфортно, ты узнал(а) много нового

4 было сложно, ты не знал(а) ничего из материала

5 было слишком сложно, многое осталось для тебя непонятным



Домашнее задание



До встречи!