

## Семинар 3

### 1. Задача 3 (из семинара 2)

3. Produce the Pascal's triangle up to 10th rows, using explicit and recurrence formulas for the binomial coefficients. Check the result by expanding the expression  $(x + y)^{10}$ .

Явное задание:

```
In[1]:= Column[Table[Binomial[n, k], {n, 0, 10}, {k, 0, n}], Center]
|колонка |табл... |биномиальный коэффициент |центр
```

```
Out[1]=
      {1}
    {1, 1}
  {1, 2, 1}
{1, 3, 3, 1}
{1, 4, 6, 4, 1}
{1, 5, 10, 10, 5, 1}
{1, 6, 15, 20, 15, 6, 1}
{1, 7, 21, 35, 35, 21, 7, 1}
{1, 8, 28, 56, 70, 56, 28, 8, 1}
{1, 9, 36, 84, 126, 126, 84, 36, 9, 1}
{1, 10, 45, 120, 210, 252, 210, 120, 45, 10, 1}
```

Проверка:

```
In[2]:= (x + y)^10 // Expand
|раскрыть скобки
```

```
Out[2]= x^10 + 10 x^9 y + 45 x^8 y^2 + 120 x^7 y^3 + 210 x^6 y^4 + 252 x^5 y^5 + 210 x^4 y^6 + 120 x^3 y^7 + 45 x^2 y^8 + 10 x y^9 + y^10
```

Рекуррентное задание:

```
In[3]:= l1 = {1, 3, 3, 1}
```

```
Out[3]= {1, 3, 3, 1}
```

```
In[4]:= Table[l1[[i]] + l1[[i + 1]], {i, 1, Length[l1] - 1}]
|таблица значений |длина
```

```
Out[4]= {4, 6, 4}
```

Соберем все в функцию:

```
In[5]:= l[n_] :=
  Append[Prepend[Table[l[n - 1][[i]] + l[n - 1][[i + 1]], {i, 1, Length[l[n - 1]] - 1}], 1], 1]
|добав... |добавит... |таблица значений |длина
```

```
In[6]:= l[0] = {1}
```

```
Out[6]= {1}
```

```
In[7]:= l[1] = {1, 1}
```

```
Out[7]= {1, 1}
```

```
In[8]:= l[2]
```

```
Out[8]= {1, 2, 1}
```

```
In[9]:= l[3]
```

```
Out[9]= {1, 3, 3, 1}
```

```
In[13]:= l[8]
```

```
Out[13]= {1, 8, 28, 56, 70, 56, 28, 8, 1}
```

Видим что функция медленно работает .

Рекуррентная функция “с памятью”:

```
In[14]:= ll[n_] := ll[n] = Append[
```

```
    |добавить в конец
```

```
    Prepend[Table[ll[n - 1][[i]] + ll[n - 1][[i + 1]], {i, 1, Length[ll[n - 1]] - 1}], 1], 1]
```

```
    |добавит... |таблица значений
```

```
    |длина
```

```
In[15]:= ll[0] = {1}; ll[1] = {1, 1};
```

Сравним время работы :

```
In[16]:= ll[8] // AbsoluteTiming
```

```
    |длительность по настенным часам
```

```
Out[16]= {0.0002192, {1, 8, 28, 56, 70, 56, 28, 8, 1}}
```

```
In[17]:= l[8] // AbsoluteTiming
```

```
    |длительность по настенным часам
```

```
Out[17]= {4.25405, {1, 8, 28, 56, 70, 56, 28, 8, 1}}
```

```
In[18]:= Column[Table[ll[n], {n, 0, 10}], Center]
```

```
    |колонка |таблица значений
```

```
    |центр
```

```

      {1}
    {1, 1}
  {1, 2, 1}
{1, 3, 3, 1}
{1, 4, 6, 4, 1}
{1, 5, 10, 10, 5, 1}
{1, 6, 15, 20, 15, 6, 1}
{1, 7, 21, 35, 35, 21, 7, 1}
{1, 8, 28, 56, 70, 56, 28, 8, 1}
{1, 9, 36, 84, 126, 126, 84, 36, 9, 1}
{1, 10, 45, 120, 210, 252, 210, 120, 45, 10, 1}
```

## 2. Задача 4 (из семинара 2)

```
In[ ]:= 4. Show that the number of odd numbers among the first 450 Fibonacci numbers is twice as large
as number of even numbers.
```

Проверки на нечетность/четность:

```
In[19]:= OddQ[2]
```

```
    |нечётное число?
```

```
Out[19]= False
```

```
In[20]:= EvenQ[2]
         |чѐтное число?
```

```
Out[20]= True
```

Число Фибоначчи :

```
In[21]:= Fibonacci[4]
         |число Фибоначчи
```

```
Out[21]= 3
```

Построим список из 450 элементов :

```
In[22]:= s = Table[Fibonacci[i], {i, 1, 450}];
         |табл... |число Фибоначчи
```

Посчитаем количество четных и нечетных :

```
In[23]:= {Length@Select[s, OddQ], Length[Select[s, EvenQ]]}
         |длина |выбрать |нечѐт... |длина |выбрать |чѐтное числ
```

```
Out[23]= {300, 150}
```

Проверим утверждение из задания :

```
In[24]:= Length@Select[s, OddQ] == 2 Length[Select[s, EvenQ]]
         |длина |выбрать |нечѐтное... |длина |выбрать |чѐтное чи
```

```
Out[24]= True
```

### 3. Задача 1 (из семинара 3)

1. Find the local extrema and max & min values of the function

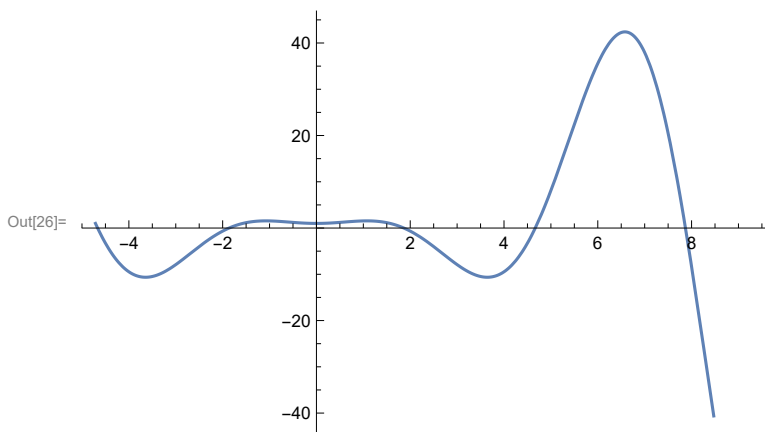
```
In[ ]:=
```

$$f(x) = x^2 \cos(x) + 1,$$

where  $x \in [-3\pi/2, 3\pi]$ . Make illustration.

```
In[25]:= f[x_] := x^2 Cos[x] + 1
         |косинус
```

```
In[26]:= Plot[f[x], {x, -3 π / 2, 3 π}]
         |график функции
```



Локальные минимумы :

In[27]:= **m1 = FindMinimum[f[x], {x, -2}]**  
 [найти минимум]

Out[27]=  $\{-10.6378, \{x \rightarrow -3.6436\}\}$

In[28]:= **m2 = FindMinimum[f[x], {x, 2}]**  
 [найти минимум]

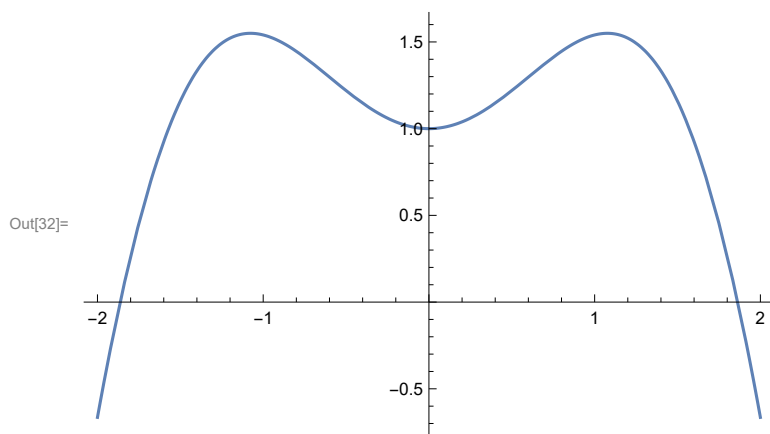
Out[28]=  $\{-10.6378, \{x \rightarrow 3.6436\}\}$

In[29]:= **m3 = FindMinimum[f[x], {x, 0}]**  
 [найти минимум]

FindMinimum: Encountered a gradient that is effectively zero. The result returned may not be a minimum; it may be a maximum or a saddle point.

Out[29]=  $\{1., \{x \rightarrow 0.\}\}$

In[32]:= **Plot[f[x], {x, -2, 2}]**  
 [график функции]



Локальные максимумы :

In[31]:= **mm1 = FindMaximum[f[x], {x, 6}]**  
 [найти максимум]

Out[31]=  $\{42.4032, \{x \rightarrow 6.57833\}\}$

In[33]:= **mm2 = FindMaximum[f[x], {x, 1}]**  
 [найти максимум]

Out[33]=  $\{1.54977, \{x \rightarrow 1.07687\}\}$

In[34]:= **mm3 = FindMaximum[f[x], {x, -1}]**  
 [найти максимум]

Out[34]=  $\{1.54977, \{x \rightarrow -1.07687\}\}$

Наименьшее и наибольшее значение функции на заданном отрезке :

In[35]:= **min = Minimize[{f[x],  $-3\pi/2 \leq x \leq 3\pi$ }, x]**  
 [минимизировать]

Out[35]=  $\{1 - 9\pi^2, \{x \rightarrow 3\pi\}\}$

```
In[36]:= max = NMaximize[{f[x], -3 π / 2 ≤ x ≤ 3 π}, x]
```

[численная максимизация](#)

```
Out[36]:= {42.4032, {x → 6.57833}}
```

Генерация точек вида (x, y) из результатов команд на примере минимума m1:

```
In[37]:= m1
```

```
Out[37]:= {-10.6378, {x → -3.6436}}
```

```
In[38]:= {x /. m1[[2]], m1[[1]]}
```

```
Out[38]:= {-3.6436, -10.6378}
```

Визуализация:

```
In[39]:= Plot[{f[x], min[[1]], max[[1]]}, {x, -3 π / 2, 3 π},
```

[график функции](#)

```
  AxesLabel → {"x", "f(x)"}, AxesStyle → Arrowheads[0.02], Epilog →
```

[обозначения на осях](#)

[стиль осей](#)

[наконечники](#)

[эпилог](#)

```
  {LightGray, Rectangle[{1, -80}, {5, -40}], Opacity[.4], Green, PointSize[0.02],
```

[светло-се...](#) [прямоугольник](#)

[прозрачность](#)

[зелё...](#) [размер точки](#)

```
  Point[{x /. m1[[2]], m1[[1]]}, {x /. m2[[2]], m2[[1]]}, {x /. m3[[2]], m3[[1]]}, {2, -50}],
```

[точка](#)

```
  Red, Point[{x /. mm1[[2]], mm1[[1]]}, {x /. mm2[[2]], mm2[[1]]},
```

[красная точка](#)

```
  {x /. mm3[[2]], mm3[[1]]}, {2, -70}], Opacity[1], Black,
```

[прозрачность](#) [чёрный](#)

```
  Text["- Local min", {3.5, -50}], Text["- Local max", {3.5, -70}]
```

[текст](#)

[текст](#)

```
}, PlotLegends →
```

[легенды графика](#)

```
{ "f(x)", "Min Value=" <> ToString[N@min[[1]], "Max Value=" <> ToString[max[[1]]]}
```

[минимум](#)

[преобраз...](#)

[численное пр...](#)

[максимум](#)

[преобразовать в строку](#)

