### Решение задач

#### Задача 1

Даны три целых числа. Возвести в квадрат отрицательные числа и в третью степень — положительные (число 0 не изменять).

```python

def transform\_numbers(a, b, c):

numbers = [a, b, c]

result = []

for num in numbers:

if num < 0:

result.append(num \*\* 2)

elif num > 0:

result.append(num \*\* 3)

else:

result.append(num)

return result

print(transform\_numbers(-2, 0, 3)) # Output: [4, 0, 27]

```

#### Задача 2

Из трех данных чисел выбрать наименьшее и наибольшее.

```python

def min\_and\_max(a, b, c):

numbers = [a, b, c]

return min(numbers), max(numbers)

print(min\_and\_max(-2, 0, 3)) # Output: (-2, 3)

```

#### Задача 3

Ввести с клавиатуры два числа. Определить что больше: сумма квадратов или квадрат суммы этих чисел. Ответ вывести в виде сообщения.

```python

def compare\_sums(x, y):

sum\_of\_squares = x\*\*2 + y\*\*2

square\_of\_sum = (x + y)\*\*2

if sum\_of\_squares > square\_of\_sum:

return "Сумма квадратов больше."

elif sum\_of\_squares < square\_of\_sum:

return "Квадрат суммы больше."

else:

return "Они равны."

print(compare\_sums(3, 4)) # Output: "Квадрат суммы больше."

```

#### Задача 4

Ввести с клавиатуры значения трех сторон треугольника a, b и c и определить является ли он прямоугольным. Ответ вывести в виде сообщения.

```python

def is\_right\_triangle(a, b, c):

sides = sorted([a, b, c])

if sides[0]\*\*2 + sides[1]\*\*2 == sides[2]\*\*2:

return "Треугольник прямоугольный."

else:

return "Треугольник не прямоугольный."

print(is\_right\_triangle(3, 4, 5)) # Output: "Треугольник прямоугольный."

```

#### Задача 5

Найти периметр и площадь прямоугольного треугольника если даны длины его катетов a и b.

```python

def triangle\_properties(a, b):

hypotenuse = (a\*\*2 + b\*\*2) \*\* 0.5

perimeter = a + b + hypotenuse

area = 0.5 \* a \* b

return perimeter, area

print(triangle\_properties(3, 4)) # Output: (12.0, 6.0)

```

#### Задача 6

Дана длина окружности. Найти площадь круга ограниченного этой окружностью. В качестве значения Pi использовать 3.14.

```python

def circle\_area(circumference):

pi = 3.14

radius = circumference / (2 \* pi)

area = pi \* radius\*\*2

return area

print(circle\_area(31.4)) # Output: 78.5

```

#### Задача 7

Даны натуральные числа x и y. Вычислить произведение используя лишь операцию сложения. Задачу решить двумя способами.

```python

def multiply\_using\_addition(x, y):

# Способ 1

result = 0

for \_ in range(y):

result += x

return result

def multiply\_using\_recursion(x, y):

if y == 0:

return 0

return x + multiply\_using\_recursion(x, y - 1)

print(multiply\_using\_addition(3, 4)) # Output: 12

print(multiply\_using\_recursion(3, 4)) # Output: 12

```

#### Задача 8

Даны два целых числа A и B (A < B). Вывести все целые числа расположенные между данными числами (включая сами эти числа) в порядке их возрастания, а также количество N этих чисел.

```python

def range\_and\_count(A, B):

numbers = list(range(A, B + 1))

count = len(numbers)

return numbers, count

print(range\_and\_count(3, 7)) # Output: ([3, 4, 5, 6, 7], 5)

```

#### Задача 9

В заданной строке посчитать количество русских букв «А».

```python

def count\_russian\_A(s):

return s.count('А')

print(count\_russian\_A("ААБВАА")) # Output: 4

```

#### Задача 10

В заданной строке удалить второй и четвертый по счету символы.

```python

def remove\_characters(s):

return s[:1] + s[2:3] + s[4:]

print(remove\_characters("Пример")) # Output: "Пимер"

```

#### Задача 11

Класс `Persona` содержит информацию о ФИО человека, дате рождения и адресе человека. Метод подсчитывает количество дней оставшихся до следующего дня рождения. По заданному ФИО человека узнать количество дней оставшихся до следующего дня рождения.

```python

from datetime import datetime, timedelta

class Persona:

def \_\_init\_\_(self, fio, birth\_date, address):

self.fio = fio

self.birth\_date = datetime.strptime(birth\_date, "%Y-%m-%d")

self.address = address

def days\_until\_birthday(self):

today = datetime.today()

next\_birthday = self.birth\_date.replace(year=today.year)

if today > next\_birthday:

next\_birthday = next\_birthday.replace(year=today.year + 1)

return (next\_birthday - today).days

# Пример использования:

persona = Persona("Иван Иванов", "2000-07-15", "Москва")

print(persona.days\_until\_birthday()) # Вывод зависит от текущей даты

```

#### Задача 12

Напишите метод заменяющий в строке все вхождения слова «бяка» на «вырезано цензурой».

```python

def censor\_word(s):

return s.replace("бяка", "вырезано цензурой")

print(censor\_word("Это полная бяка!")) # Output: "Это полная вырезано цензурой!"

```

#### Задача 13

Дан массив размера N. Вывести его элементы в обратном порядке.

```python

def reverse\_array(arr):

return arr[::-1]

print(reverse\_array([1, 2, 3, 4, 5])) # Output: [5, 4, 3, 2, 1]

```

#### Задача 14

Создайте массив содержащий 10 первых нечетных чисел. Выведите элементы массива на консоль в одну строку разделяя запятой.

```python

def odd\_numbers\_array():

return [i for i in range(1, 20, 2)]

print(", ".join(map(str, odd\_numbers\_array()))) # Output: "1, 3, 5, 7, 9, 11, 13, 15, 17, 19"

```

#### Задача 15

a) Tkinter: Создать форму произвольного размера

б) На этой форме разместить следующие элементы:

a) Button

b) Label

c) Entry

в) Каждый элемент имеет свою цветовую характеристику (фон/текст)

г) Обработать нажатие или выбор для каждого элемента, выдав соответствующее сообщение.

```python

import tkinter as tk

from tkinter import messagebox

def on\_button\_click():

messagebox.showinfo("Info", "Button clicked!")

def create\_form():

root = tk.Tk()

root.title("Форма")

button = tk.Button(root, text="Кнопка", bg="blue", fg="white", command=on\_button\_click)

button.pack(pady=10)

label = tk.Label(root, text="Метка", bg="green", fg="black")

label.pack(pady=10)

entry = tk.Entry(root, bg="yellow", fg="black")

entry.pack(pady=10)

root.mainloop()

create\_form()

```

#### Задача 16

Дана строка S и число N. Преобразовать строку S в строку длины N следующим образом: если длина строки S больше N, то отбросить первые символы, если длина строки S меньше N, то в ее начало добавить символы "." (точка).

```python

def adjust\_string\_length(s, n):

if len(s) > n:

return s[-n:]

return '.' \* (n - len(s)) + s

print(adjust\_string\_length("пример", 10)) # Output: "....пример"

print(adjust\_string\_length("пример", 3)) # Output: "мер"

```

#### Задача 17

Построить строку, состоящую из малых букв латин

ского алфавита (по алфавиту).

```python

def latin\_alphabet():

return "".join(chr(i) for i in range(ord('a'), ord('z') + 1))

print(latin\_alphabet()) # Output: "abcdefghijklmnopqrstuvwxyz"

```

#### Задача 18

TkinterГрафика: Рисуем линии, прямоугольники, круг и текст в Tkinter.

```python

import tkinter as tk

def draw\_shapes():

root = tk.Tk()

root.title("Графика")

canvas = tk.Canvas(root, width=400, height=400, bg="white")

canvas.pack()

# Линия

canvas.create\_line(50, 50, 200, 50, fill="blue", width=2)

# Прямоугольник

canvas.create\_rectangle(50, 100, 200, 150, outline="red", width=2)

# Круг (окружность)

canvas.create\_oval(50, 200, 150, 300, outline="green", width=2)

# Текст

canvas.create\_text(100, 350, text="Пример текста", fill="purple", font=("Helvetica", 16))

root.mainloop()

draw\_shapes()

```

### Android

#### 1. Создаем кнопки в Android.

Создание кнопки в Android требует разработки на Java или Kotlin. Пример на Kotlin:

```kotlin

import android.os.Bundle

import android.widget.Button

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val button = findViewById<Button>(R.id.myButton)

button.setOnClickListener {

// Действие при нажатии кнопки

}

}

}

```

#### 2. Написать видеоплеер

```kotlin

import android.net.Uri

import android.os.Bundle

import android.widget.MediaController

import android.widget.VideoView

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val videoView = findViewById<VideoView>(R.id.videoView)

val mediaController = MediaController(this)

mediaController.setAnchorView(videoView)

val uri = Uri.parse("android.resource://" + packageName + "/" + R.raw.sample\_video)

videoView.setMediaController(mediaController)

videoView.setVideoURI(uri)

videoView.requestFocus()

videoView.start()

}

}

```

#### 3. Написать аудиоплеер

```kotlin

import android.media.MediaPlayer

import android.os.Bundle

import android.widget.Button

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

private lateinit var mediaPlayer: MediaPlayer

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

mediaPlayer = MediaPlayer.create(this, R.raw.sample\_audio)

val playButton = findViewById<Button>(R.id.playButton)

val pauseButton = findViewById<Button>(R.id.pauseButton)

val stopButton = findViewById<Button>(R.id.stopButton)

playButton.setOnClickListener {

mediaPlayer.start()

}

pauseButton.setOnClickListener {

if (mediaPlayer.isPlaying) {

mediaPlayer.pause()

}

}

stopButton.setOnClickListener {

if (mediaPlayer.isPlaying) {

mediaPlayer.stop()

mediaPlayer.prepare()

}

}

}

override fun onDestroy() {

super.on уничтожить()

if (mediaPlayer.isPlaying){

mediaPlayer.stop()

mediaPlayer.release()

}

}

}

#### 4. Изменить картинку по нажатию кнопки

import android.os.Bundle

import android.widget.Button

import android.widget.ImageView

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val imageView = findViewById<ImageView>(R.id.imageView)

val button = findViewById<Button>(R.id.button)

button.setOnClickListener {

imageView.setImageResource(R.drawable.new\_image) // замените new\_image на ресурс новой картинки

}

}

}

```

#### 5. ArrayAdapter

```kotlin

import android.os.Bundle

import android.widget.ArrayAdapter

import android.widget.ListView

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val listView = findViewById<ListView>(R.id.listView)

val items = arrayOf("Item 1", "Item 2", "Item 3")

val adapter = ArrayAdapter(this, android.R.layout.simple\_list\_item\_1, items)

listView.adapter = adapter

}

}

```

#### 6. Телефонные звонки

Для реализации звонков необходимо добавить разрешение в `AndroidManifest.xml`:

```xml

<uses-permission android:name="android.permission.CALL\_PHONE"/>

```

Код для совершения звонка:

```kotlin

import android.content.Intent

import android.net.Uri

import android.os.Bundle

import android.widget.Button

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val callButton = findViewById<Button>(R.id.callButton)

callButton.setOnClickListener {

val phoneIntent = Intent(Intent.ACTION\_CALL)

phoneIntent.data = Uri.parse("tel:123456789")

startActivity(phoneIntent)

}

}

}

```

#### 7. Android RatingBar

```kotlin

import android.os.Bundle

import android.widget.RatingBar

import android.widget.Toast

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val ratingBar = findViewById<RatingBar>(R.id.ratingBar)

ratingBar.setOnRatingBarChangeListener { ratingBar, rating, fromUser ->

Toast.makeText(this, "Rating: $rating", Toast.LENGTH\_SHORT).show()

}

}

}

```

#### 8. Google Maps (для Алматы ключ - AIzaSyDlrysRab7buK9aufWAvtt\_HbOFAGI0KnE)

```kotlin

import android.os.Bundle

import androidx.appcompat.app.AppCompatActivity

import com.google.android.gms.maps.CameraUpdateFactory

import com.google.android.gms.maps.GoogleMap

import com.google.android.gms.maps.OnMapReadyCallback

import com.google.android.gms.maps.SupportMapFragment

import com.google.android.gms.maps.model.LatLng

import com.google.android.gms.maps.model.MarkerOptions

class MapsActivity : AppCompatActivity(), OnMapReadyCallback {

private lateinit var mMap: GoogleMap

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_maps)

val mapFragment = supportFragmentManager

.findFragmentById(R.id.map) as SupportMapFragment

mapFragment.getMapAsync(this)

}

override fun onMapReady(googleMap: GoogleMap) {

mMap = googleMap

val almaty = LatLng(43.238949, 76.889709)

mMap.addMarker(MarkerOptions().position(almaty).title("Marker in Almaty"))

mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(almaty, 10f))

}

}

```

#### 9. Аналоговые часы

```kotlin

import android.os.Bundle

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

}

}

```

Добавьте в `activity\_main.xml` виджет часов:

```xml

<AnalogClock

android:id="@+id/analogClock"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerInParent="true"/>

```

#### 10. Android - Image Effects

```kotlin

import android.graphics.BitmapFactory

import android.os.Bundle

import android.widget.ImageView

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val imageView = findViewById<ImageView>(R.id.imageView)

val bitmap = BitmapFactory.decodeResource(resources, R.drawable.sample\_image)

// Примените здесь эффекты к изображению

imageView.setImageBitmap(bitmap)

}

}