Program do zadania Nr.7:

Wzór trapezów i metoda Romberga:

```python
import numpy as np
import math


def trapez(f, a, b, n):
    h = (b - a) / n
    x = a
    In = f(a)
    for k in range(1, n):
        x = x + h
        In += 2 * f(x)

    K = (In + f(b)) * h * 0.5
    print("Trapezoid: ", round(K, 7))
    return K


def romberg(f, a, b, p):
    I = np.zeros((p, p))
    for i in range(0, p):
        I[i, 0] = trapez(f, a, b, 2 ** i)
        for j in range(0, i):
            I[i, j + 1] = (4 ** (j + 1) * I[i, j] - I[i - 1, j]) / (4 ** (j + 1) - 1)
        x = (I[i, 0:i + 1])
        print_row(x)

    return I


def print_row(X):
    print("Romberg: ", ' | '.join('%11.8f' % x for x in X))


def func(x):
    return math.sin(math.pi * (1 + math.sqrt(x)) / (1 + math.pow(x, 2))) * math.pow(math.e, -x)


if __name__ == '__main__':
    p_rows = 20
    I = romberg(func, 0, math.log(10) * 7, p_rows)
    solution = I[p_rows - 1, p_rows - 1]
    print("Final solution: ", round(solution, 7))
```

Wyniki do zadania Nr.2:

Wyniki metody Romberga i wzoru trapezów znajdują się w plikach tekstowych:
1. Results_Romberg.txt
2. Results_TrapezRule.txt

Komentarze do zadania Nr.7:

W tym zadaniu dla obliczania całki

$$I = \int_0^\infty \sin\left(\pi \frac{1+\sqrt{x}}{1+x^2}\right) e^{-x} \mathrm{d}x$$

było skorzystano ze wzoru trapezów oraz metody Romberga.
Najpierw obliczamy górną granicę A z nierówności:

$$e^{-A} < 10^{-7}$$

Wynik: A = 7 * ln 10

Metoda Romberga:

Metoda Romberga łączy wzór trapezów z ekstrapolacją Richardsona. Poniżej znajduje się przegląd procesu całkowania:

Let $R_{i-1} = I_i$

1. Starts with the computation of one panel and two panels via Composite Trapezoidal Rule

$$R_{1,1} = I_1 \ (\textit{one panel})$$
$$R_{2,1} = I_2 \ (\textit{two panels})$$

2. We get the leading error term $C_1 h^2$ then eliminate by Richardson Extrapolation using p = 2 (since that is the exponent of the error)

$$R_{2,2} = \frac{2^2 R_{2,1} - R_{1,1}}{2^2 - 1} = \tfrac{4}{3} R_{2,1} - \tfrac{1}{3} R_{1,1}$$

3. Get the leading error term $C_2 h^4$ then eliminate by Richardson Extrapolation using p = 4
4. We do this until it converges to the wanted solution

$$\begin{tikzcd}
R_{1,1} \arrow[dr] & & & \\
R_{2,1} \arrow[r] \arrow[dr] & R_{2,2} \arrow[dr] & & \\
R_{3,1} \arrow[r] & R_{3,2} \arrow[r] & R_{3,3} & \\
\vdots & \vdots & \vdots & \ddots \\
R_{n,1} \arrow[r] & R_{n,2} \arrow[r] & R_{n,3} \quad \cdots & R_{n,n}
\end{tikzcd}$$

To mówi nam, że musimy obliczyć, skąd pochodzą dwie strzałki, aby obliczyć, gdzie wskazują dwie strzałki. Najdokładniejszym oszacowaniem całki jest zawsze ostatni człon ukośny tablicy. Proces ten jest kontynuowany, aż różnica między dwoma kolejnymi wyrazami ukośnymi stanie się wystarczająco mała.

Powyższy przegląd można podsumować wzorem:

General Romberg Formula

$$R_{i,j} = \begin{cases} R_{i,j} = CTR(h_i) & : \ \text{if } i = 1 \\ R_{i,j} = \dfrac{4^{j-1}R_{i,j-1} - R_{i-1,\,j-1}}{4^{j-1}-1} & : \text{if } i>1 \end{cases}$$

*note: $h = \frac{b-a}{2^{i-1}}$: *where a and b are the boundaries*

: *i is the index of the row*

CTR oznacza wzór trapezów.

Wzór trapezów:

Metoda trapezów

Dużo lepszym rozwiązaniem jest zastosowanie trapezów o wysokości *dx* i podstawach równych odpowiednio wartości funkcji w punktach krańcowych.

Sama zasada nie zmienia się. Przedział całkowania $<x_p, x_k>$ dzielimy na $n+1$ równo odległych punktów $x_0, x_1, x_2, \ldots, x_n$. Punkty te wyznaczamy w prosty sposób wg wzoru:

$$x_i = x_p + \frac{i}{n}(x_k - x_p) \quad dla \quad i = 1,2,\ldots,n$$