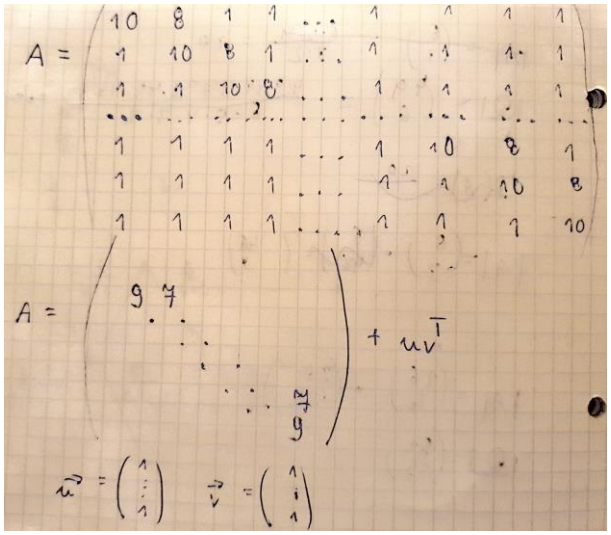
Zadanie NUM4

Aby uruchomić, należy wpisać w konsoli make run. Program został napisany w języku python(3.9.9).

Aby znaleźć szukany wektor, podaną macierz zapisuję jako sumę macierzy A o elemencie diagonalnym 9 i nad diagonalnym 7, oraz wektory jedynek. Pozwala to wykorzystać strukturę macierzy A i wykonywać obliczenia tylko na tych elementach. Dzięki temu uzyskana zostaje mniejsza złożoność obliczeniowa w stosunku do sytuacji, kiedy zadeklarowana byłaby pełna macierz.



W programie wykorzystałem bibliotekę numpy, aby zainicjować puste macierze w których następnie w pętli umieszczam odpowiednie elementy oraz aby dostosować jej wymiary i przedstawić wyniki w odpowiedniej formie.

Do obliczeń zaimplementowałem algorytm backward substitution (funkcja solve_upper), ponieważ macierz A jest macierzą trójkątną górną.

Wyniki:

Wektor y:

[[0.07525844, 0.07525904, 0.07525827, 0.07525926, 0.07525799, 0.07525963,

0.07525752, 0.07526023, 0.07525674, 0.07526122, 0.07525546, 0.07526287,

0.07525334, 0.07526559, 0.07524985, 0.07527009, 0.07524406, 0.07527753,

0.0752345, 0.07528983, 0.07521869, 0.07531015, 0.07519256, 0.07534375,

0.07514936, 0.07539929, 0.07507795, 0.0754911, 0.07495991, 0.07564287,

0.07476477, 0.07589376, 0.0744422, 0.07630849, 0.07390898, 0.07699406,

0.07302753, 0.07812736, 0.07157043, 0.08000077, 0.06916176, 0.08309763,

0.06518009, 0.08821693, 0.05859813, 0.09667944, 0.04771776, 0.11066849,

0.02973183, 0.13379325]]