# Математичний модуль

Останнім часом у світі проявляється тенденція використовувати мову програмування python для розробки програмних модулів, що мають справу із складними математичними обчисленнями.

Це пов’язано із бажанням дослідників заощадити час на розробці власне програмного забезпечення і зосередитися на математичних проблемах, які вони розв’язують.

Такі заощадження досягаються за рахунок використання великої кількості вже готових програмних модулів, таких як numpy.linalg для роботи з лінійною алгеброю та scipy.integrate для чисельного інтегрування.

У своєму модулі ми будемо опиратися на ці наробки минулих поколінь дослідників і використовувати їх у повній мірі.

Ми робимо наш програмний модуль відкритим для громадськості. Він доступний онлайн за [посиланням](https://github.com/Sky-Nik/mathematical-modelling). Ми будемо раді будь-яким змістовним доповненням до нього, а також виправленням помилок, якщо такі там раптом присутні.

## Власне програмний модуль

Перелік процедур, реалізованих у програмному модулі включно із коротким описом:

* **solve\_system** – розв’язує (можливо несумісну) систему лінійних алгебраїчних рівнянь, тобто покриває підрозділ 1.1 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L8).
* **solve\_summed\_system** – розв’язує лінійну дискретно-підсумовану систему рівнянь. Разом із наступною процедурою покриває підрозділ 1.2 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L21).
* **solve\_time\_summed\_system** – розв’язує лінійну дискретно-підсумовану систему рівнянь, де окремі значення трактуються як значення певної функції часової змінної у певні дискретні моменти часу. Разом із попередньою процедурою покриває підрозділ 1.2 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L38).
* **solve\_distributed\_system** – розв’язує лінійну дискретно-розподілену систему рівнянь. Разом із наступною процедурою покриває підрозділ 1.3 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L58).
* **solve\_time\_distributed\_system** – розв’язує лінійну дискретно-розподілену систему рівнянь, де окремі значення трактуються як значення певної функції часової змінної у певні дискретні моменти часу. Разом із попередньою процедурою покриває підрозділ 1.3 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L76).
* **solve\_integral\_system** – розв’язує лінійну інтегрально-перетворювальну систему рівнянь. Разом із наступною процедурою повністю покриває розділ 1.4 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L97).
* **solve\_functional\_system** – розв’язує лінійну функціонально-перетворювальну систему рівнянь. Разом із попередньою процедурою повністю покриває розділ 1.4 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L124).
* **solve\_1d\_space\_distributed\_integral\_system** – розв’язує розподілену у просторі інтегрально-перетворювальну систему рівнянь у одновимірному випадку. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L152).
* **solve\_2d\_space\_distributed\_integral\_system** – розв’язує розподілену у просторі інтегрально-перетворювальну систему рівнянь у двовимірному випадку. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L229). Разом із наступними двома процедурами ці дві процедури повністю покривають розділ 1.5 посібника. Не зважаючи на необхідність чисельного обчислення потрійного інтегралу працює відносно швидко, за долі секунди.
* Процедура **solve\_3d\_space\_distributed\_integral\_system** не була реалізована через необхідність чисельного обчислення чотири-кратного інтегралу. Похибка подібних обчислень була б катастрофічною. Окрім цього немає зручного способу візуалізації отриманого розв’язку. І, що найголовніше, подібні задачі не мають фізичного підґрунтя.
* **solve\_1d\_space\_distributed\_functional\_system** – розв’язує розподілену у просторі функціонально-перетворювальну систему рівнянь у одновимірному випадку. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L315).
* **solve\_2d\_space\_distributed\_functional\_system** – розв’язує розподілену у просторі функціонально-перетворювальну систему рівнянь у двовимірному випадку. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L360). Разом із попередніми двома процедурами ці дві процедури повністю покривають розділ 1.5 посібника. Не зважаючи на необхідність чисельного обчислення потрійного інтегралу працює відносно швидко, за долі секунди.
* Процедура **solve\_3d\_space\_distributed\_functional\_system** не була реалізована через необхідність чисельного обчислення чотири-кратного інтегралу. Похибка подібних обчислень була б катастрофічною. Окрім цього немає зручного способу візуалізації отриманого розв’язку. І, що найголовніше, подібні задачі не мають фізичного підґрунтя.
* **solve\_1d\_discrete\_observations\_discrete\_modelling** – розв’язує дискретно-спостережувану систему у обмеженій просторово-часовій області із одновимірною за простором областю. Використовує дискретні моделюючі функції. Разом із наступною процедурою повністю покриває підрозділ 2.2.1 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L410).
* **solve\_2d\_discrete\_observations\_discrete\_modelling** – розв’язує дискретно-спостережувану систему у обмеженій просторово-часовій області із двовимірною за простором областю. Використовує дискретні моделюючі функції. Разом із попередньою процедурою повністю покриває підрозділ 2.2.1 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L505).
* Процедура **solve\_3d\_discrete\_observations\_discrete\_modelling** не була реалізована через надмірну громіздкість коду.
* **solve\_1d\_discrete\_observations\_continuous\_modelling** – розв’язує дискретно-спостережувану систему у обмеженій просторово-часовій області із одновимірною за простором областю. Використовує неперервні моделюючі функції. Разом із наступною процедурою повністю покриває підрозділ 2.3.1 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L608). Не зважаючи на необхідність чисельного обчислення великої кількості інтегралів працює відносно швидко, за долі секунди.
* **solve\_2d\_discrete\_observations\_continuous\_modelling** – розв’язує дискретно-спостережувану систему у обмеженій просторово-часовій області із двовимірною за простором областю. Використовує неперервні моделюючі функції. Разом із попередньою процедурою повністю покриває підрозділ 2.3.1 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L744). Зважаючи на необхідність чисельного обчислення великої кількості подвійних інтегралів працює відносно повільно, може займати до однієї хвилини за великої кількості спостережень.
* Процедура **solve\_3d\_discrete\_observations\_continuous\_modelling** не була реалізована через необхідність чисельного обчислення великої кількості потрійних інтегралів, що могло б зайняти десятки хвилин, якщо не години обчислень. Окрім цього похибка таких обчислень була б катастрофічною. І, що найголовніше, подібні задачі не мають фізичного підґрунтя.
* **solve\_1d\_continuous\_observations\_discrete\_modelling** – розв’язує неперервно-спостережувану систему у обмеженій просторово-часовій області із одновимірною за простором областю. Використовує дискретні моделюючі функції. Разом із наступною процедурою повністю покриває підрозділ 2.4.1 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L891). Не зважаючи на необхідність чисельного обчислення великої кількості інтегралів працює відносно швидко, за долі секунди.
* **solve\_2d\_continuous\_observations\_discrete\_modelling** – розв’язує неперервно -спостережувану систему у обмеженій просторово-часовій області із двовимірною за простором областю. Використовує дискретні моделюючі функції. Разом із попередньою процедурою повністю покриває підрозділ 2.4.1 посібника. Повна документація процедури доступна за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/lib.py#L1031). Зважаючи на необхідність чисельного обчислення великої кількості подвійних інтегралів працює відносно повільно, може займати до однієї хвилини за великої кількості спостережень.
* Процедура **solve\_3d\_continuous\_observations\_discrete\_modelling** не була реалізована через необхідність чисельного обчислення великої кількості потрійних інтегралів, що могло б зайняти десятки хвилин, якщо не години обчислень. Окрім цього похибка таких обчислень була б катастрофічною. І, що найголовніше, подібні задачі не мають фізичного підґрунтя.

## Тестування

Враховуючи, що повний код програмного модуля займає 1187 рядків коду, було б неприпустимим покладатися на відсутність жодної друкарської помилки у ньому.

Як наслідок, було написано велику кількість різноманітних тестів, які перевіряють різні властивості розв’язків, які видають реалізовані нами процедури.

Зокрема тестується:

* відповідність знайденого розв’язку точному аналітичному розв’язку для модельних задач для яких він існує;
* обмеженість похибки розв’язку за умов, що точного розв’язку задача на має;
* швидкодія складних процедур на великих вхідних даних, таких як сотні моделюючих точок і десятки умов, що накладаються на задачу.

У зв’язку із таким строгим і всебічним тестування, сам тестовий модуль займає 818 рядків. Але на нього ми вже можемо покластися, адже він не містить у собі ніякої складної логіки.

Наприклад тестування задачі із дискретними спостереженнями і дискретними моделюючими функціями у одновимірному випадку виглядає так:

# задаємо точки для початкових  
cond\_x0s\_list = [0.0, 1.0]  
# і крайових умов   
cond\_xtGammas\_list = [  
 (0.0, 0.5), (0.0, 1.0),  
 (1.0, 0.5), (1.0, 1.0),  
]  
  
# задаємо значення умов у початкових  
cond\_f0s\_list = [1.0, 1.0]  
# і крайових точках  
cond\_fGammas\_list = [  
 1.0, 1.0,  
 1.0, 1.0,  
]  
  
# моделюючі точки для гладкої частини розв’язку  
model\_xtInfs\_list = [  
 (0.0, 0.0), (0.5, 0.0), (1.0, 0.0),  
 (0.0, 0.5), (0.5, 0.5), (1.0, 0.5),  
 (0.0, 1.0), (0.5, 1.0), (1.0, 1.0),  
]  
# початкові моделюючі точки  
model\_x0s\_list = [0.0, 0.5, 1.0]  
# крайові моделюючі точки  
model\_xtGammas\_list = [  
 (0.0, 0.0), (1.0, 0.0),  
 (0.0, 0.5), (1.0, 0.5),  
 (0.0, 1.0), (1.0, 1.0),  
]  
  
# функція правої частини рівняння  
def f(*x*: *float*, *t*: *float*) -> *float*:  
 return 1.0  
  
# функція «Гріна» оператора (сам оператор можна не задавати)  
def g(*x*: *float*, *t*: *float*) -> *float*:  
 return 1.0  
  
# аналітичний розв’язок  
def desired(*x*: *float*, *t*: *float*) -> *float*:  
 return 1.0  
  
# отриманий нами розв’язок  
actual = lib.solve\_1d\_discrete\_observations\_discrete\_modelling(  
 cond\_x0s\_list, cond\_xtGammas\_list, cond\_f0s\_list, cond\_fGammas\_list,  
 model\_xtInfs\_list, model\_x0s\_list, model\_xtGammas\_list, f, g)  
  
# точки для звірки аналітичного і нашого розв’язків  
xts\_list = [  
 (0.0, 0.0), (0.5, 0.0), (1.0, 0.0),  
 (0.0, 0.5), (0.5, 0.5), (1.0, 0.5),  
 (0.0, 1.0), (0.5, 1.0), (1.0, 1.0),  
]  
  
# Звіряємо рівність аналітичного і обчисленого нами розв’язків  
for x\_i, t\_i in xts\_list:  
 np.testing.assert\_almost\_equal(desired(x\_i, t\_i), actual(x\_i, t\_i))

Тестовий модуль містить багато прикладів використання реалізованих нами процедур і доступний за [посиланням](https://github.com/Sky-Nik/mathematical-modelling/blob/master/tests.py).