KIMYO INTERNATIONAL UNIVERSITY IN CENTRAL ASIAN JOURNAL OF STEM
ISSN 2181-2934 http://stem.kiut.uz/

OPENSEES DASTURI BILAN TANISHUV

O.Mirzahmedov[1,a)], K. Abdivakhidov[1,b)]

[1.] *Toshkent Kimyo Xalqaro Universiteti*

[a)] e-mail: otabek.mirzakhmedov@gmail.com
[b)] e-mail: kamol.abdivakhidov@gmail.com

Annotatsiya. OpenSees(Open System for Earthquake Engineering Simulation) konstruksiya va geotexnik inshootlarni dinamikasini simulatsiya qiluvchi ochiq dasturiy ta'minot. Hozirda ushbu dastur konstruksiyalarni olov, shamol, zilzila kabi turli tabiiy ofatlardagi harakatini chekli elementlar yordamida taxlil qiladi. Ushbu maqolada OpenSees dasturi bilan tanishib unda sodda konstruksiyani ko'rib chiqamiz va OpenSees qanday ishlashi haqida tushunchaga ega bo'lamiz. Uning boshqa shu kabi dasturlardan farqi va ustunliklari taxlil qilinadi.

Kalit so'zlar: OpenSees, Python, konstruksiya simulatsiyasi

Annotation. OpenSees (Open System for Earthquake Engineering Simulation) is an open software that simulates the dynamics of structures and geotechnical structures. Currently, this program analyzes the behavior of structures in various natural disasters such as fire, wind, and earthquake using finite elements. In this article, we will get acquainted with the OpenSees program, look at its simple construction and get an understanding of how OpenSees works. Its differences and advantages from other similar programs are analyzed.

Keywords: OpenSees, Python, structure simulation

Аннотация. OpenSees (Open System for Earthquake Engineering Simulation) — это открытое программное обеспечение, которое моделирует динамику структур и геотехнических сооружений. В настоящее время эта программа анализирует поведение конструкций при различных стихийных бедствиях, таких как пожар, ветер и землетрясение, с использованием конечных элементов. В этой статье мы познакомимся с программой OpenSees, посмотрим на ее простую конструкцию и поймем, как работает OpenSees. Анализируются ее отличия и преимущества от других подобных программ.

Ключевые слова: OpenSees, Python, моделирование конструкции.

Kirish

Bino va inshootlarni zilzila va boshqa tabiiy ofatlarga chidamli qilib loyihalash va tekshirishda simulatsiya yoki matematik modellashtirish muxim rol o'ynaydi. Simulatsiya - modellarni tez o'zgartirish imkonini beruvchi dastur yoki dasturiy to'plamni talab etadi. OpenSees 1990-yillarda Pacific Earthquake Engineering Research(PEER)ilmiy tekshirish instituti tomonidan bino va inshootlar zilzila bardoshligini simulatsiya qilish uchun ishlab chiqilgan. OpenSees eng so'nggi chekli element modellar(FEM) va chiziqsiz hisoblash algoritmlarini(nonlinear computing algorithms) o'z ichiga oladi. U asosan C++ va Fortran dasturlash tillarida yozilgan va mutlaqo bepul [1-3].

OpenSees foydalanuvchidan dasturlash ko'nikmalarini talab qiladi va asosan kod skriptlari yozish orqali ishlaydi. Bu foydalanuchiga dasturlash ichidagi sikl, xolatni tekshirish(if-else) kabi konstruktlardan foydalanish imkonini berib, foydalanuvchi bir necha modellarni tekshirish va optimal yechimni topa olishi mumkin [2].

OpenSeesda modellash

OpenSeesda sodda konstruksiyani modellab ko'ramiz va bu bizga OpenSees haqida ko'proq tushunchaga ega bo'lishga yordam beradi. OpenSeesni python dasturlash tili yordamida yozilgan OpenSeespy kutubxonasidan foydalanamiz. Python o'rganishga qulay va oson bo'lgani uchun hozirda ko'plab ilmiy loyixalarda ishlatilmoqda. Pythonni turli xil dasturlash muhitlarida(IDE) ishlatish mumkin va ushbu maqolada Jupyter Notebook tanlangan [4-5].

Dastlab ish OpenSeespy kutubxonalarini yuklashdan boshlanadi (Rasm 1). Rasm1da Pythonning import buyrug'i orqali OpenSeesdan tashqari yana 2ta kutubxona yuklangan ular modellarni grafigini chizish uchun ishlatiladi.

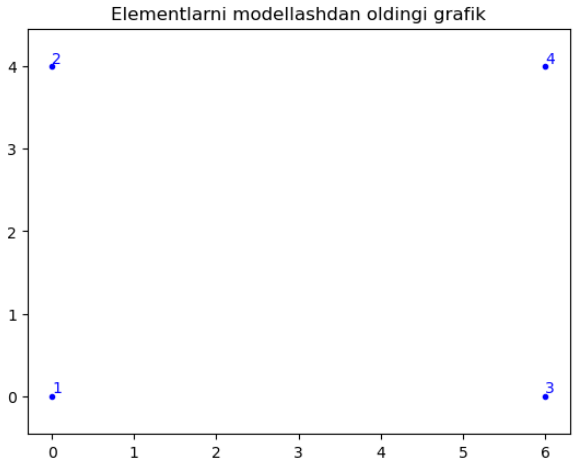
Rasm 1. Kutubxonalarni yuklash

Keyingi bosqich modelashtirish bo'lib unda dastab Model tanladi (Rasm 2). Rasm 2da ko'rsatilganidek 'basic' ya'ni sodda model tanlangan va ndm yonida 2 turibdi bu degani 2Dda ishlaniladi va ndf bu erkinlik darajasi bo'lib va 3tanlangan ya'ni har bir element gorizontal va vertikal yo'nalishlarda ko'cha oladi va aylana oladi. Undan so'ng ustun va to'sin uzunligi, elastik modulus, ko'ndalang kesim yuzlari va inersiya momentlari kiritilgan. Node buyrug'i yordamida nuqtalar raqami va ularning koordinatasini kiritilgan. Fix buyrug'i orqali qotirishlar amalga oshirilagan.

```
ops.wipe()
ops.model('basic', '-ndm', 2, '-ndf', 3)

UstunUzunligi, BalkaUzunligi = 4., 6.

AUstun, ABalka = 2.e-3, 6.e-3 # ustun va to'sin ko'ndalang kesim yuzasi
IzUstun, IzBalka = 1.6e-5, 5.4e-5  #ustun va to'sin inersiya momentlari

E = 200.e9 # elastik modulus

Ep = {1: [E, AUstun, IzUstun],
      2: [E, AUstun, IzUstun],
      3: [E, ABalka, IzUstun]}

ops.node(1, 0., 0.)
ops.node(2, 0., UstunUzunligi)
ops.node(3, BalkaUzunligi, 0.)
ops.node(4, BalkaUzunligi, UstunUzunligi) # nuqtalrni kiritish

ops.fix(1, 1, 1, 1) #qattiq qotirish
ops.fix(3, 1, 1, 0) # sharnirli qotirish

opsv.plot_model()
plt.title('Elementlarni modellashdan oldingi grafik')
```

Rasm 2. Nuqtalarni kiritish

Rasm 2dagi kod yozilgandan so’ng yurg’izsak quyidagi natijani olamiz.

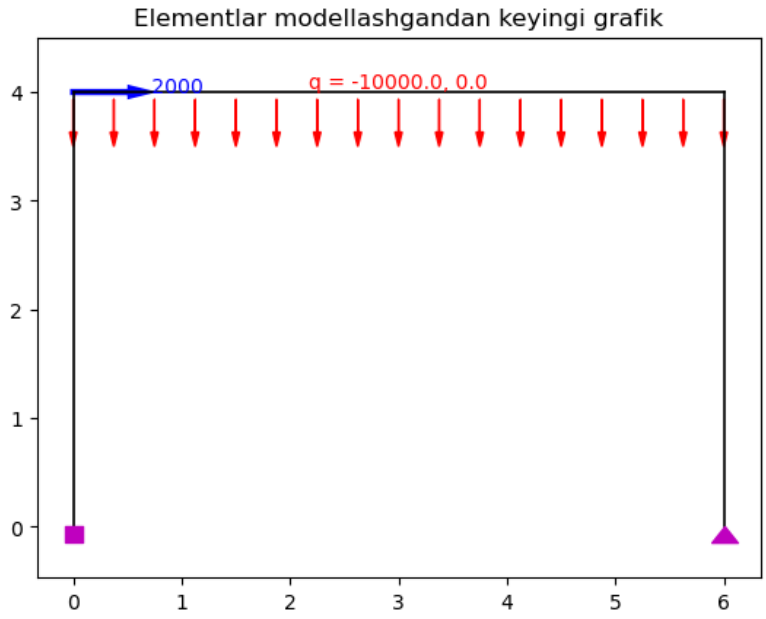


Rasm 3. Kiritilgan nuqtalar joylashuvi

Nuqtalar joylashtirilib chiqqandan so’ng elementlarni modellash boshlanadi. Elementlarni element buyrug’i orqali kiritiladi, dastlab uning turi masalan bizni modelda ‘elasticBeamColumn’ ya’ni elastik ustun va to’sin tanlangan. Agar siz fermalar bilan ishlasangiz ‘truss’ni tanlashingiz kerak va xakozo. Element turidan so’ng uni raqami va qaysi nuqtalar birlashuvidan yasalagan uning yuzasi va boshqa ma’lumotlar kiritiladi. Undan so’ng yuklar kiritilaga ‘load’ va ‘eleload’ buyruqlari orqali. Tahlil turi sifatida Statik(static) tanlangan va algoritm esa chiziqli(linear) tanlangan.

```
ops.geomTransf('Linear', 1)
# ustunlar
ops.element('elasticBeamColumn', 1, 1, 2, AUstun, E, IzUstun, 1)
ops.element('elasticBeamColumn', 2, 3, 4, AUstun, E, IzUstun, 1)
# To'sin
ops.element('elasticBeamColumn', 3, 2, 4, ABalka, E, IzBalka, 1)

Px = 2.e+3 # bir nuqtaga jamlangan yuk
Wy = -10.e+3
Wx = 0.

Ew = {3: ['-beamUniform', Wy, Wx]}

ops.timeSeries('Constant', 1)
ops.pattern('Plain', 1, 1)
ops.load(2, Px, 0., 0.)

for etag in Ew:
    ops.eleLoad('-ele', etag, '-type', Ew[etag][0], Ew[etag][1],
                Ew[etag][2])

ops.constraints('Transformation')
ops.numberer('RCM')
ops.system('BandGeneral')
ops.test('NormDispIncr', 1.0e-6, 6, 2)
ops.algorithm('Linear')
ops.integrator('LoadControl', 1)
ops.analysis('Static')
ops.analyze(1)

ops.printModel()

opsv.plot_model()
plt.title('Elementlar modellashgandan keyingi grafik')

opsv.plot_loads_2d()

# sfac = 80.

opsv.plot_defo()
```

Rasm 4. Elementlarni modellash



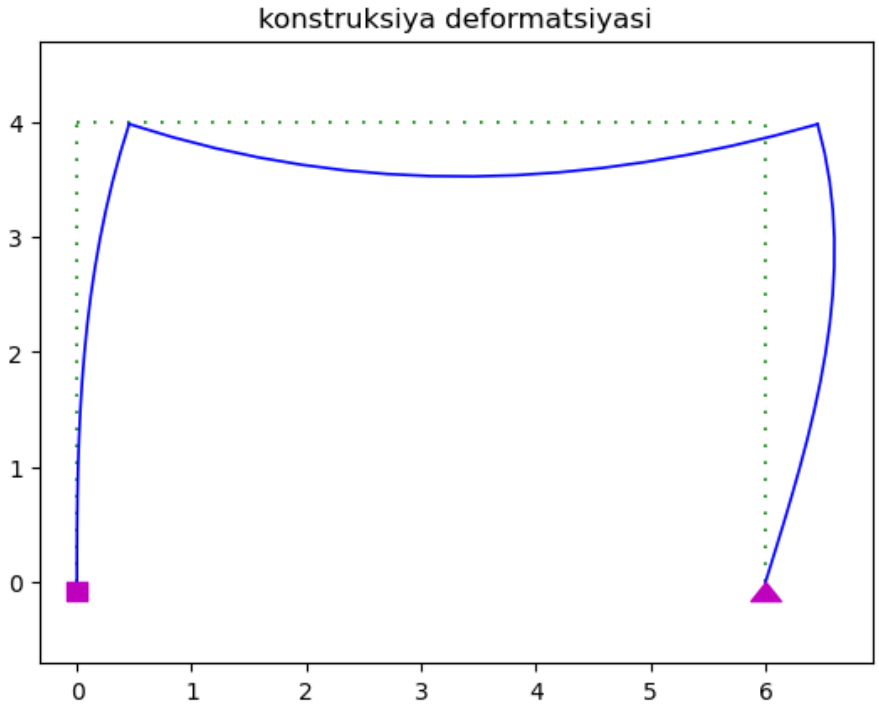
Rasm 5. Elementlarni modellashdan keyingi grafik

Elementlarni kiritib va taxlil turlarini tanlaganimizdan so’ng natijalarni grafik shaklda olish uchun quyidagi buyruqlarni kiritamiz.

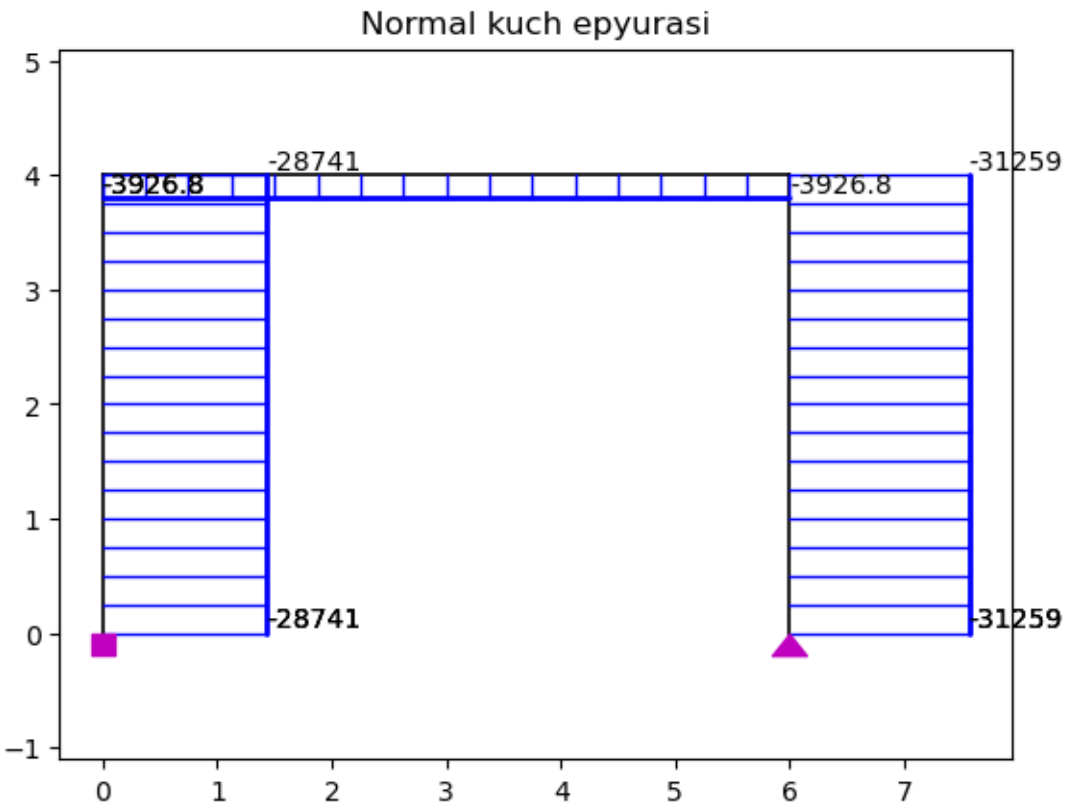
```
opsv.plot_defo()
plt.title('konstruksiya deformatsiyasi')
# opsv.plot_defo(sfac)
# fmt_interp = {'color': 'blue', 'linestyle': 'solid', 'linewidth': 1.2, 'marker': '.', 'markersize': 6}
# opsv.plot_defo(sfac, fmt_interp=fmt_interp)

# 4. plot N, V, M forces diagrams

sfacN, sfacV, sfacM = 5.e-5, 5.e-5, 5.e-5

opsv.section_force_diagram_2d('N', sfacN)
plt.title('Normal kuch epyurasi')

opsv.section_force_diagram_2d('T', sfacV)
plt.title('Urinma kuch epyurasi')

opsv.section_force_diagram_2d('M', sfacM)
plt.title('Moment epyurasi')

plt.show()

exit()
```

Rasm 6. Natijalarni grafik shaklda olish buyruqlari

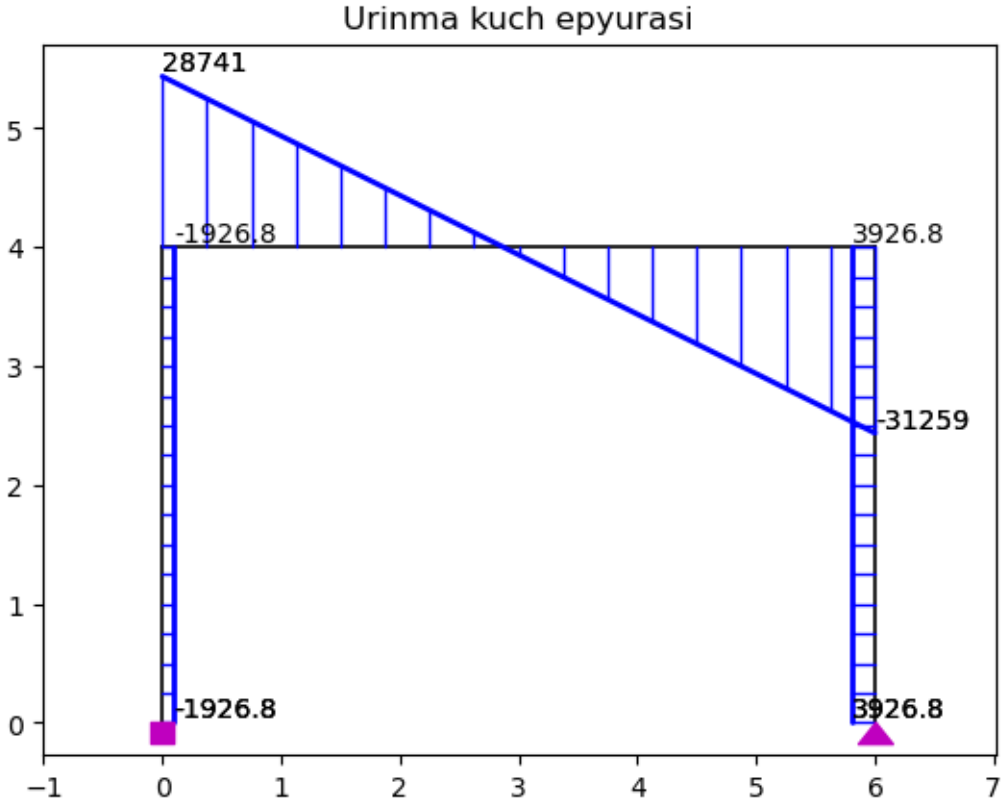
Natijalar quyidagi grafiklarda ko’rsatilagan.



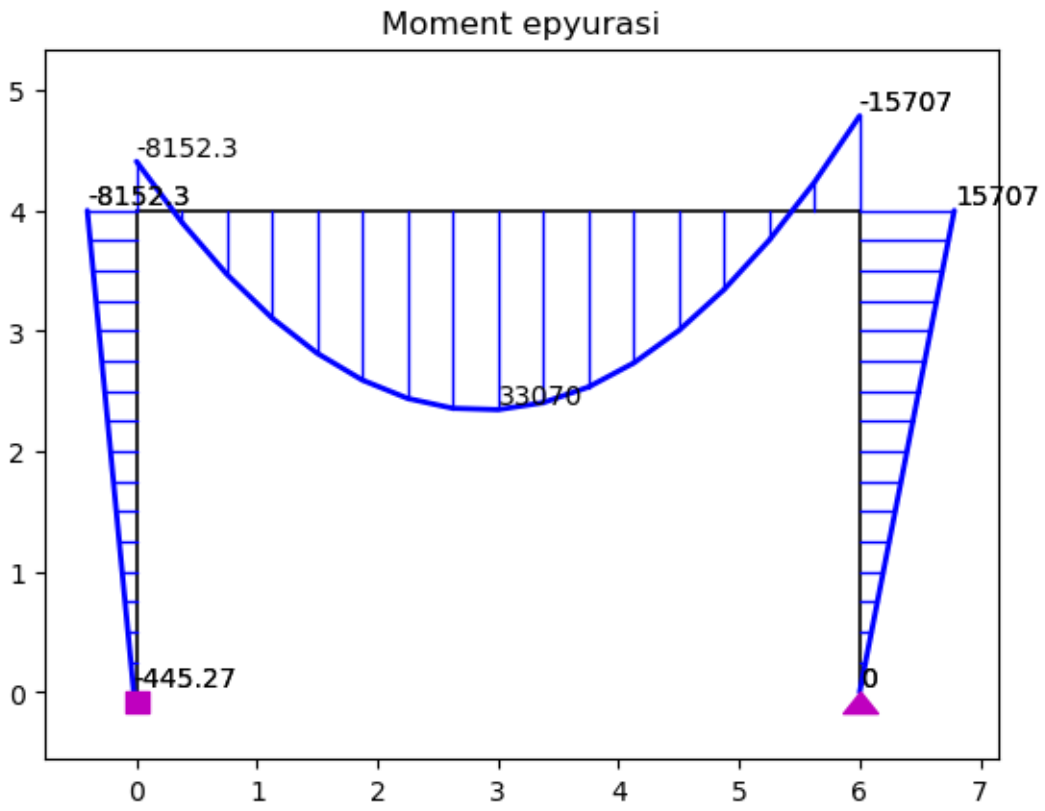
Rasm 7. Konstruksiya deformatsiyasi



Rasm 8. Normal kuch epyurasi



Rasm 9. Urinma kuch epyurasi



Rasm 10. Moment epyurasi

Xulosa

OpenSeesda biz yuqorida ko'rib chiqqan modeldan murakkab bo'lgan konstruksiyalarni ham taxlil qilsa bo'ladi. OpenSees asosan chiziqsiz (nonlinear) taxlillar uchun ishlatilinadi. Bu kabi taxlillarni Sap2000, ETABs yoki Lira kabi dasturlar yordamida qilsa bo'ladi. Lekin ular bepul emas, OpenSees esa bepul lekin dasturlash ko'nikmasini talab qiladi. Sap2000, ETABs yoki Lira kabi dasturlarda modelning istalgan bir qismida uzunlik yoki boshqa parametri o'zgaradigan bo'lsa uni boshqatdan qilishga to'g'ri keladi. OpenSeasda esa biz ko'rganimizdek bu juda oson.

Bu kabi dasturlarni talabalarga o'rgatish juda ham foydali bo'lib bu ularni faqatgina pullik dasturlarga bog'lanib qolishdan asraydi. Va nazariyaning muhimligini tushinib yetishadi. Buning uchun ularga Python va Matlab kabi dasturlarni o'rgatish kerak.

Foydalanilgan adabiyotlar

1. McKenna, Frank. "OpenSees: a framework for earthquake engineering simulation." *Computing in Science & Engineering* 13.4 (2011): 58-66
2. Zhu, Minjie, Frank McKenna, and Michael H. Scott. "OpenSeesPy: Python library for the OpenSees finite element framework." *SoftwareX* 7 (2018): 6-11.
3. Elhami Khorasani, Negar, Maria EM Garlock, and Spencer E. Quiel. "Modeling steel structures in OpenSees." *Computers and Structures* 157.C (2015): 218-231.
4. https://opensees.berkeley.edu/
5. https://openseespydoc.readthedocs.io/en/latest/