

# Moodle Virtual Programming Lab (VPL) дээр Тооцон бодох физик хичээлийн дасгал үүсгэж бэлдэх нь

Б.Алтанхүү

*МУИС, ШУС, Физикийн Тэнхим*

## Хураангуй

Энэхүү өгүүлэлд, сургалт удирдах Moodle цахим системийн Virtual Programming Lab (VPL) автомат үнэлгээний хэрэгслийн тусламжтайгаар, “Тооцон бодох физик” (Computational Physics) хичээлийн хүрээнд програм бичих дасгал үүсгэж бэлдэх, дасгалын гүйцэтгэлийг автоматаар үнэлэх боломжуудыг цөмийн физикийн Резерфордын сарнилын бодлогын жишээ дээр авч үзлээ. Энэ нь урьдчилсан шатны судалгаа бөгөөд зөвхөн багшийн зүгээс хийж гүйцэтгэх ёстой бэлтгэл (техник) ажлыг тоймлон дурдсан. Moodle ба VPL хэрэгслийг сургалтын үйл ажиллагаанд бодитоор хэрэглэхэд ямар үр дүн гарч буйг нарийвчлан авч үзэх нь судалгааны ажлын дараачийн үе шат байх болно. **Түлхүүр үгс:** Сургалтын технологи, сургалтын хэрэглэгдэхүүн, автомат үнэлгээний систем, цөмийн физик, мэдээллийн технологи, e-learning, Python.

## I. УДИРТГАЛ

Сургалт удирдах Moodle (Modular Object-Oriented Dynamic Learning Environment) систем нь хичээл сургалтын явцыг удирдан хөтлөх боломж олгодог, нээлттэй эх бүхий үнэгүй цахим талбар (програм хангамж) юм. Өнөөдөр дэлхий даяар олон зуун сая хэрэглэгч сургалтын болон бусад зориулалтаар уг системийг ашиглаж байгаа бөгөөд цаашид энэ тоо улам өсөх хандлагатай [1]. Moodle нь сүлжээний орчинд client-server дизайны дагуу ажилладаг. Веб сервер дээр Moodle-ийг байршуулж (moodle server), харин жирийн хэрэглэгчид Firefox, Chrome, Edge гэх мэт түгээмэл веб хөтөч ашиглан онлайнаар системтэй холбогдоно (moodle clients). Moodle нь Google Forms-той харьцуулахад олон дахин илүү боловсронгуй, сургалт удирдах платформ.

Moodle цахим орчинд, програмчлалын автомат үнэлгээний **Virtual Programming Lab (VPL)** хэрэгслийн (plug-in) (Зураг 1) тусламжтайгаар, Python, Octave/Matlab, C/C++, Java, Javascript, Maple, R, VHDL гэх мэт түгээмэл ба мэргэжлийн програмчлалын хэлүүдээр код бичиж, бичсэн кодыг ажиллуулж, автоматаар шалгаж дүгнэх боломжтой байдаг [1, 2].



Зураг 1. Virtual Programming Lab.

[3]-т дурдсанаар бол, коллежийн түвшинд мэдээллийн технологийн мэргэжлийн суурь хичээлүүдийг заахдаа Moodle ба VPL-ийг хослуулан ашиглаж туршсан үр дүн

урьдчилсан байдлаар эерэг (**програмчлалын олон өөр хэл дээр код бичиж болдог**; тухайн хэрэгслүүд **үнэгүй бас нээлттэй эх бүхий гэдэг** нь тухайн мэргэжлээр суралцдаг оюутны тоо өсөж буй үед **өртөг зардлын хувьд хэмнэлттэй**, бүрэн дүүрэн ашиглаж чадвал **оюутны гүйцэтгэлийг үнэлэх багшийн ажлыг тодорхой хэмжээнд автоматжуулан ачаалал хөнгөлнө**) гэжээ. Нөгөө талаас, **багшийн зүгээс бэлтгэл ажиллагаа ихтэй өөрөөр хэлбэл VPL дээр дасгал бэлдэж тохиргоог нь хийх үе шатанд багш үлэмж цаг-хөдөлмөр зарцуулж** байна гэдгийг тэмдэглэсэн байна.

Тэгвэл [4]-т, их сургуулийн түвшинд мэдээллийн технологийн мэргэжлийн суурь хичээлүүдийг судалж буй 37 оюутанд Moodle VPL ашиглан туршилтын хичээл орж; мөн тухайн чиглэлээр хичээл заадаг, Moodle ба VPL-ийг олон жилийн туршид хэрэглэж ирсэн туршлагатай 7 багшийг хамруулсан байна. Тэд VPL ашиглахын давуу талыг, өмнөх [3]-т дурдсан дээр нэмээд, **Moodle системтэй холбогддог** (оюутнууд, багш нар), дасгалын гүйцэтгэлийг **бусдаас хуулсан эсэхийг шалгах (integrated plagiarism checking) боломжтой** байдаг (багш нар), **бичиж илгээсэн кодыг автоматаар шалгаад зөв бурууг тэр дор нь хариу мэдээлэх шуурхай гэдрэг холбоотой (automatic feedback)** (оюутнууд) гэж дурдсан бол, дутагдалтай тал нь **анх удаа тулж ажиллахад бэрхшээлтэй** байдал (оюутнууд, багш нар) гэсэн байна. VPL-ийн автомат гэдрэг холбоо нь **хүссэн газраасаа багшийн оролцоогүйгээр бие даан дасгалыг ажиллах боломж олгож буйг** оюутнууд эерэгээр дүгнэжээ [5].

[6]-д, бакалаврын түвшинд програмчлалын хичээлийг VPL ашиглан өрнүүлэх нь код бичиж сурах шинэ арга барилыг оюутанд санал болгож улмаар сурч танин мэдэх дур сонирхлыг өдөөсөн эерэг нөлөө үзүүлж байна гэж урьдчилсан байдлаар дүгнэхийн зэрэгцээ хангалттай урт удаан хугацааны туршид илүү олон оюутан, багшийг хамарч судалгааг явуулах шаардлагатай хэмээн дурдсан байна.

[7]-д, 2017-2021 оны хооронд хэвлэгдсэн, сургалтын зориулалттай програмчлалын автомат үнэлгээний (VPL, CodeRunner, Travis-CI гэх мэт) системүүдийн талаарх 121 судалгааны ажлыг тоймлон авч үзсэн байна. Ихэнх ийм системд, өмнө дурдсан давуу талуудаас гадна гүйцэтгэсэн дасгалаа олон удаа давтан илгээх боломжтой учир амжилттай дүн авах магадлалыг нэмэгдүүлж байгаа гэж оюутнууд эерэгээр үнэлсэн ч гэдгэр холбоог илүү боловсронгуй буюу ойлгомжтой болгох шаардлагатай гэсэн нь өмнөх судалгаануудын үр дүнтэй давхцаж байна. Түүнчлэн мэдээллийн технологийн цоо шинэхэн үзэгдэл - хиймэл оюуны систем (Large Language Models) нь компьютерын шинжлэх ухааны сургалтад нөлөөлөхүйц түвшинд хүрсэн тухай бас [7]-д дурджээ. Тухайлбал ChatGPT-ийн сүүлийн үеийн хувилбар нь, энгийн ойлгомжтой тодорхой даалгавар (prompt) өгсөн тохиолдолд бүрэн ажиллагаатай зөв програмын код гаргаж өгөх чадвартай болсон байна. Энэ мэт нөхцөл байдлыг цаашид анхаарч үзэх шаардлагатай болсон гэжээ.

Физикийн тэнхимийн бакалаврын сургалтын төлөвлөгөөнөөс хамааран мэргэжлийн суурь (3 багц цаг) эсвэл мэргэжлийн сонгох (2 багц цаг) хичээл болдог “Тооцон бодох физик” курс дээр оюутан нь, оролт/гаралт, салаалалт, давталт бүхий энгийн програм зохиохоос эхлээд зарим хялбар тоон аргатай танилцаж тооцооны програм бичих үйл ажиллагаанд суралцдаг. Үүгээрээ тус курс нь компьютерын шинжлэх ухаан, мэдээллийн технологийн мэргэжлийн суурь хичээлүүдтэй төстэй учир Moodle (эсвэл адил зориулалттай өөр бусад) цахим орчин болон VPL (эсвэл адил зориулалттай өөр бусад) хэрэгслийг ашиглан хичээлийг өрнүүлэх боломжтой гэж үзэж байна. Өрнүүлэх аргачлалыг дараах байдлаар томъёолж болох юм. Үүнд:

1. Багш: Moodle VPL дээр програм бичих дасгалыг үүсгэж, тестлэж, бэлдэнэ, шаардлагатай тохиргоонуудыг хийнэ;

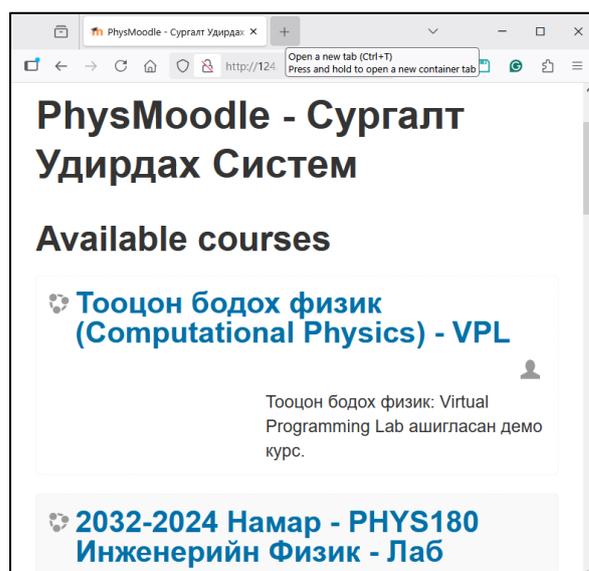
2. Оюутан: тухайн дасгалыг Moodle VPL дээр гүйцэтгэнэ (програм бичнэ);
3. Moodle VPL: оюутны гүйцэтгэлийг (бичсэн програмыг) автоматаар шалган дүгнэж (automatic assessment) зөв бурууг хариу мэдээлнэ (automatic feedback), шаардлагатай бол оноо өгнө (grading/evaluation) гэх мэт [3-7, 8, 9].

Ингэснээр, танхимд програм бичих дасгал хичээл ороход, танхимаар болон зайнаас програм бичүүлж шалгалт авахад, програм бичих гэрийн даалгавар гүйцэтгэхэд тун тохиромжтой (хичээлийн процессын зарим хэсгийг тухайлбал шалгах ба дүгнэх үйл ажиллагааг автоматжуулж багшийн ачааллыг хөнгөлсөн, оюутан хаанаас ч дасгал даалгавраа гүйцэтгэх боломжтой, багш дэргэд байхгүй байсан ч автомат гэдгэр холбоогоор дасгалын гүйцэтгэлийнхээ талаар хариу мэдээлэл авч чадна, ингэснээр оюутны идэвхийг өдөөнө) гэж үзэж байна.

## II. VIRTUAL PROGRAMMING LAB БА ЖИШЭЭ ДАСГАЛ

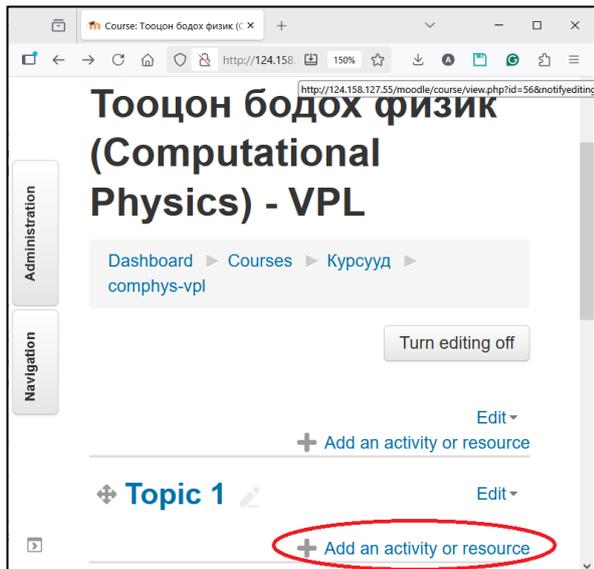
VPL хэрэгслийг (plug-in) Moodle системд хэрхэн нэмэлтээр суулгаж, үндсэн тохиргоог хийх дэлгэрэнгүй техник зааврыг [2]-оос үзэж болно.

Суулгаж тохируулсан VPL-ийг хичээлд ашиглахын тулд эхлээд харгалзах хичээл буюу курсыг Moodle дээр үүсгэсэн байх ёстой. Энэ зорилгоор, интернетэд гарцтай Ubuntu Linux үйлдлийн систем бүхий виртуал сервер түрээслэн Moodle системийг байршуулж, түүн дээрээ Тооцон бодох физик курсыг туршилтын горимоор үүсгэсэн (Зураг 2) [10].



Зураг 2. Moodle системд үүсгэсэн “Тооцон бодох физик” курс.

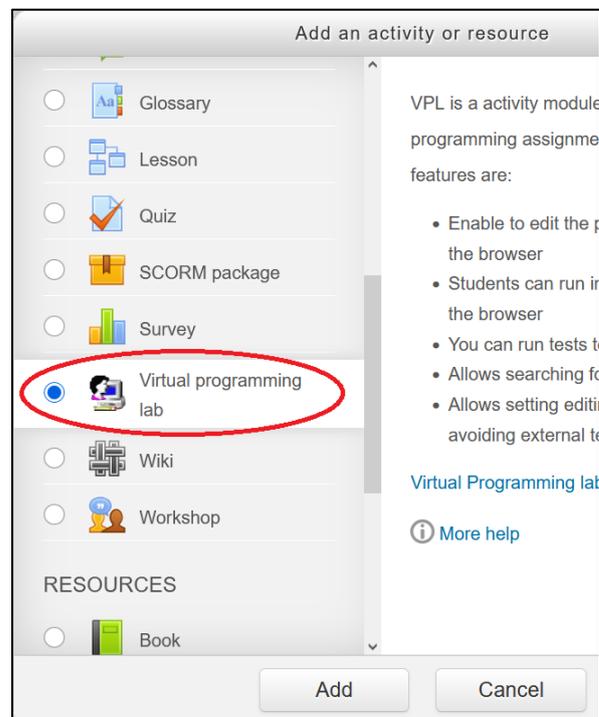
Дараа нь уг курстээ VPL-дасгал (activity) нэмнэ. Үүний тулд хичээл рүү нэвтэрч ороод Зураг 3-т тэмдэглэж үзүүлсэн командыг сонгоход Зураг 4-т харуулсан цонх гарч ирнэ.



Зураг 3. Хичээлд дасгал (activity) нэмэх.

Зураг 4-т тэмдэглэж харуулсан командыг сонгож **Add** товч дарахад Зураг 5-д үзүүлсэн цонх гарч ирнэ. Энд, үүсгэх гэж буй дасгалынхаа нэр (Name), агуулга (Full description), хүчинтэй хугацаа (Submission period), нийт оноо (Grade), давтан гүйцэтгэлд торгуулийн оноотой эсэх (Grade) гэх мэт

үндсэн тохиргоог хийнэ (Зураг 5). Дасгалыг үүсгэснийхээ дараа бусад нэмэлт тохиргоог, тухайлбал, багшийн бичсэн зөв бодолт бүхий програм (жишиг програм), оюутны бичиж оруулсан програмын хариултыг тулгаж үзэх шүүлт (test cases) зэргийг үргэлжлүүлэн тодорхойлж өгөх шаардлагатай [2, 8, 9].



Зураг 4. VPL-дасгал сонгох.

Зураг 5. VPL-дасгалын тохиргооны цонх.

Жишээ болгож, цөмийн физикийн сэдвээр тооцон бодох VPL-дасгал үүсгэе. Резерфордын сарнилын бодлогыг (альфа-бөөм хөдөлгөөнгүй алтны атомоос сарних

үзэгдлийн загварчлал) авч үзье [11]. Энэ дасгалыг ажиллах хүртэл оюутан Python хэлний үндсэн элементүүдийн талаар ойлголттой болж, тооцооны програм бичих, үр

дүнгийн график дүрслэл байгуулах анхдагч дадал чадварыг эзэмшсэн байх ёстой. Үүнд: оролт-гаралт, салаалалт, давталт, дэд програм, стандарт list ба numpy массив, SciPy модуль, Matplotlib модуль, энгийн дифференциал тэгшитгэл бодох Эйлер/Рунге-Куттын тоон аргуудтай танилцсан байх гэх мэт.

Бодлогын заавар буюу оюутанд уншигдах текстийг дараах маягаар бэлдэж болно:

1. Бодлогын үндсэн агуулгыг бичиж өгнө. Бэлэн болсон агуулга оюутанд дараах байдалтай ингэж харагдана (Зураг 6).

### Description: ode - Rutherford scattering – Euler

**Maximum number of files:** 1

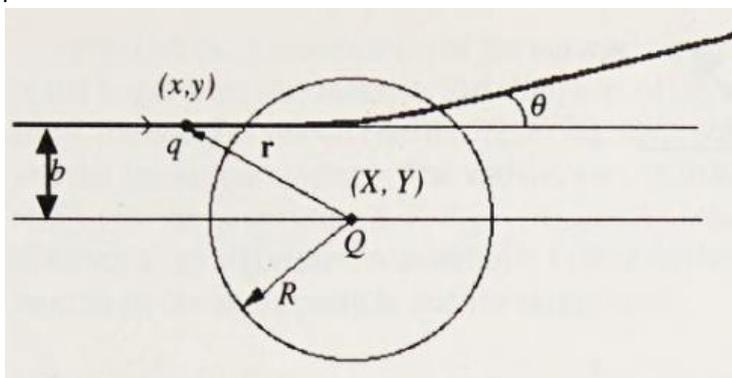
**Type of work:** Individual work

**Grade settings:** Maximum grade: 100

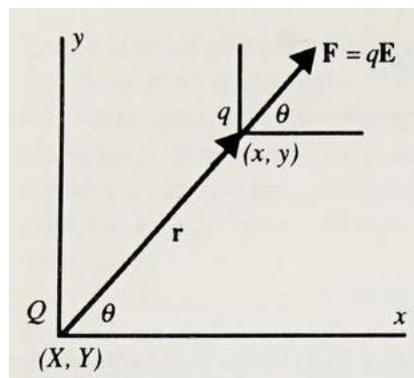
**Run:** Yes. **Evaluate:** Yes

**Automatic grade:** Yes.

Резерфордын туршилтаар альфа бөөм, хөдөлгөөнгүй алтны атомаас (цөмөөс) сарних үзэгдлийг судалдаг. Сарнилыг ХҮ хавтгай дээр авч үзье. Энд атомыг жигд түгсэн эерэг цэнэгтэй бөмбөлөг гэж үзэж байгаа.



Зураг 1.



Зураг 2.

**Бодлого.**  $(X, Y)$  цэг дээр хөдөлгөөнгүй буй  $R$  радиустай,  $M$  масстай, жигд түгсэн  $Q$  цэнэгтэй атом (цөм) руу  $m$  масстай,  $q$  цэнэгтэй бөөмийг буудав (Зураг 1). Бөөмийн анхны хурд  $v_0$ . Бөөмийн анхны байрлал  $(x_0, y_0)$ . Шагайлтын зай  $b=y_0$ . Бөөмд үйлчлэх хүчний диаграммыг Зураг 2-т харуулав. Эндээс бөөмийн

хөдөлгөөний тэгшитгэл нь  $\frac{d^2r}{dt^2} = \frac{qE}{m}$  буюу байгуулагчаар нь задалбал  $\begin{cases} \frac{d^2x}{dt^2} = \frac{qE_x}{m} \\ \frac{d^2y}{dt^2} = \frac{qE_y}{m} \end{cases}$  болно. Энд, атомын

цахилгаан орны хүчлэгийн байгуулагчууд  $E_x = \begin{cases} kQ \frac{(x-X)}{r^3}, & r > R \\ kQ \frac{(x-X)}{R^3}, & r < R \end{cases}$  ба  $E_y = \begin{cases} kQ \frac{(y-Y)}{r^3}, & r > R \\ kQ \frac{(y-Y)}{R^3}, & r < R \end{cases}$  байна.  $k =$

$\frac{1}{4\pi\epsilon_0} = 8.99 \cdot 10^{-19} \text{ Н} \cdot \text{м} \cdot \text{Кл}^{-2}$  - Кулоны тогтмол.

Алтны атомтой (цөмтэй) харилцан үйлчлэлд орж буй альфа-бөөмийн хувьд дээрх хөдөлгөөний тэгшитгэлийг Эйлерийн аргаар бодуулан бөөмийн траекторыг олж үр дүнгээ numpy массивт хадгал (эхлээд динамик list ашиглаад дараа нь numpy руу хувиргасан нь зохимжтой!). Дэлгэц дээр  $x$ ,  $y$  координатыг хугацаа  $t$ -ээс хамааруулан гурван баганаар доош цувуулж хэвлэ. Хугацааны алхам  $dt$  сек. Бодох муж  $x \in [-6, 6] \cdot 10^{-13}$  м,  $y \in [-4, 4] \cdot 10^{-13}$  м. Альфа-бөөмийн масс  $m=4$  м.а.н., энд  $1$  м.а.н.  $= 1.66 \cdot 10^{-27}$  кг - массын атом нэгж; цэнэг  $q=2e$ , энд  $e=1.60 \cdot 10^{-19}$  кл - эгэл цэнэг. Алтны атомын масс  $M=197$  м.а.н., цэнэг  $Q=79e$ .

Оролтод, атомын (цөмийн) радиус  $R$  ( $10^{-15}$ -аас  $10^{-13}$  м эрэмбийн хооронд утга), шагайлтын зай  $b$  ( $10^{-15}$ -аас  $10^{-13}$  м эрэмбийн хооронд утга байна: атомын (цөмийн) радиусаас их, радиустай ойролцоо, радиусаас бага гэх мэтээр өгөх), бодох хугацааны алхам  $dt$  ( $10^{-23}$ -аас  $10^{-20}$  с эрэмбийн хооронд утга байна) зэргийг өгнө. Хугацааны алхмыг оруулахдаа, нийт бодолтын алхмын тоо (Steps) нь хэт ихгүй (хэдэн мянга биш) байхаар баримжаалж өгөх хэрэгтэй. Гаралтад, хугацаа,  $x$  координат,  $y$  координат 3 баганаар доош цувран хэвлэгдэж, хамгийн сүүлд нь бодолтын алхмын тоо хэвлэгдэх ёстой. Бодолт, альфа бөөм зааж өгсөн  $x \in [-6, 6] \cdot 10^{-13}$  м,  $y \in [-4, 4] \cdot 10^{-13}$  м мужаас гарах хүртэл үргэлжлэн хийгдэх ёстой. Бөөмийн анхны байрлалын  $x$  координатыг бодох мужийн зүүн хил дээр,  $y$  координатыг шагайлтын зайтай тэнцүү өг. Бөөмийн анхны хурдны байгуулагчууд  $v_{0x}=2e7$  м/с (7 МэВ энергитэй),  $v_{0y}=0$  байна.

Зураг 6. Резерфордын сарнилын загварчлал (VPL-дасгал): Үндсэн агуулга.

2. Тооцоо хийдэг аливаа програм **оролт** (тооцоонд ашиглагдах өгөгдлүүд) ба **гаралттай** тооцооны үр дүн) байдаг. Оролтын өөр өөр утгад өөр өөр гаралт харгалзана. Оролт-гаралтын жишээ

утгуудыг бодлогын агуулгад нэмж оруулах нь оюутанд чиглүүлэг болох тустай. Үндсэн агуулгын араас оруулж өгнө. Доорх Зураг 7 болон Зураг 8-д үүнийг харуулжээ.

1. Жишээ нь, оролтод:

**R\_atom (m) = 1e-14**

**b (m) = 0.1e-13**

**dt (sec) = 1e-21**

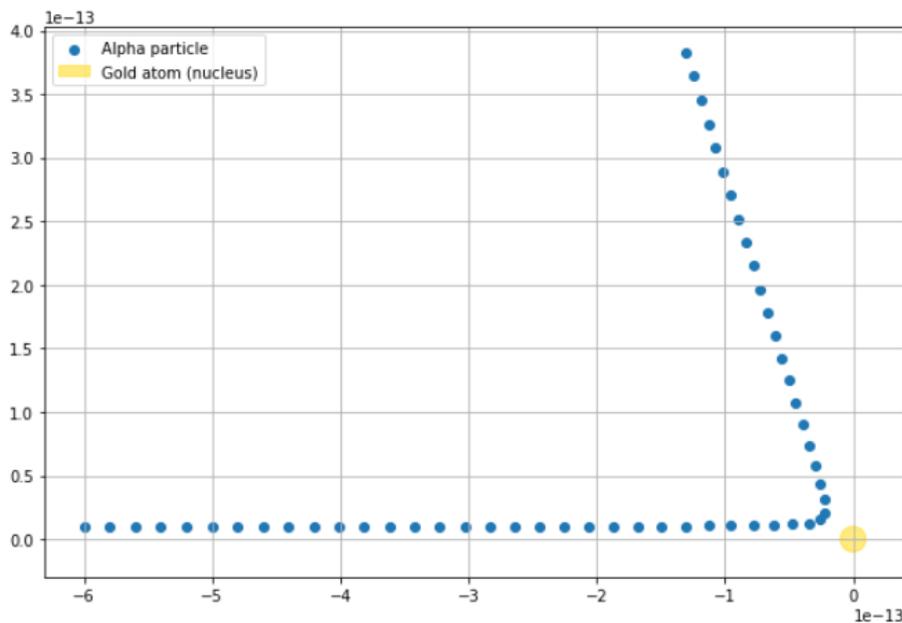
гэж өгөхөд програм:

```

t:          x:          y:
[[ 0.00000000e+00 -6.00000000e-13 1.00000000e-14]
 [ 1.00000000e-21 -5.80015206e-13 1.00002534e-14]
 [ 2.00000000e-21 -5.60046682e-13 1.00007874e-14]
 ...
 [ 5.10000000e-20 -1.18942239e-13 3.45232758e-13]
 [ 5.20000000e-20 -1.24844281e-13 3.64045069e-13]
 [ 5.30000000e-20 -1.30758318e-13 3.82892354e-13]]
Steps: 54

```

гэж гаргадаг байх ёстой. Програм 54 алхмын дараа зогссон байна. Шагайлтын зайг атомын (цөмийн) радиусаас 10 дахин их өгсөн ч, цөмийн радиус өөрөө маш бага байна. Тиймээс сайн сарниулж байна уу? Хэрэв бид энэ үр дүнгээр (Excel дээр ч юм уу) X-Y график байгуулж үзвэл иймэрхүү харагдана:



Зураг 7. Резерфордын сарнилын загварчлал (VPL-дасгал): Оролт-гаралтын жишээ 1.

Зураг 7 ба 8-аас үзэхэд, дасгалын текстийг веб браузерын стандарт html форматаар бичих, текстийн томъёо тэгшитгэлүүдийг Latex форматаар илэрхийлэх, зургийг бэлдэх гэх мэт бэлтгэл үйл ажиллагаанд багш багагүй цаг-хөдөлмөр зарцуулах нь харагдана. Түүнчлэн програмын оролт/гаралтын жишээ утгуудыг тодорхойлж өгөхийн тулд багш өөрөө юуны түрүүнд тухайн бодлогыг зөв бодсон програмын хувилбартай байх ёстой. Резерфордын сарнилын загвар мэтийн

сонгодог бодлогын хувьд бэлэн жишээ програмын кодыг (түгээмэл хэрэглэгдэгч програмчлалын хэл дээр бичсэн) харгалзах эх сурвалжуудаас олж тавин цаг хэмнэх боломжтой ч ядаж л тухайн кодыг дасгалынхаа ерөнхий формат агуулгад нийцүүлэн зүгшрүүлэхэд бас л цаг хугацаа орно.

Бэлтгэл ажлын дараагийн алхамд багш, **жишиг (зөв) хариултыг** тодорхойлж өгнө. Үүнийг **test case** гэдэг. VPL нь, оюутны бичсэн

програмын гаралтыг (хариултыг), багшийн урьдчилан бэлдсэн test case утгатай (хариулттай) тулгах замаар тухайн програмыг үнэн зөв болсон эсэхийг шалгаж чаддаг. Аль болох олон test case байвал, оюутан зөв хариултыг шууд тааж хүчээр тааруулж гаргах магадлал багасна. Манай сарнилын бодлого

загварчлах VPL-дасгалын хувьд одоохондоо 5 test case үүсгэж өгсөн (Зураг 9). Эдгээр нь оролтын ялгаатай тохиолдлуудад (атомын (цөмийн) радиусын өөр өөр утга, шагайх зайн өөр өөр утга, хугацааны алхмын өөр өөр утгад) харгалзах зөв гаралтуудыг агуулна. Test case нь оюутанд харагдахгүй байх ёстой.

2. Оролтод:

**R\_atom (m) = 1e-13**

**b (m) = 0.4e-13**

**dt (sec) = 1e-22**

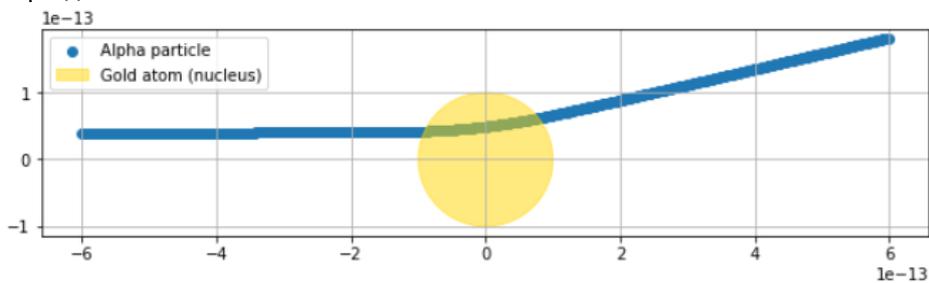
гэхэд гаралт:

```

      t:          x:          y:
[[ 0.00000000e+00 -6.00000000e-13  4.00000000e-14]
 [ 1.00000000e-22 -5.98000151e-13  4.00000101e-14]
 [ 2.00000000e-22 -5.96000454e-13  4.00000303e-14]
 ...
 [ 6.36000000e-20  5.94202694e-13  1.80659004e-13]
 [ 6.37000000e-20  5.96151096e-13  1.81116754e-13]
 [ 6.38000000e-20  5.98099632e-13  1.81574545e-13]]
Steps: 639

```

байх ёстой. Альфа бөөм 639 алхмын дараа мужийг орхиж гарсан байна. Шагайлтын зай нь цөмийн радиустай ойролцоо эрэмбийнх байгаа. Энэ тохиолдолд бөөмийн траектори дараах байдалтай харагдана:



Зураг 8. Резерфордын сарнилын загварчлал (VPL-дасгал): Оролт-гаралтын жишээ 2.

```

63 ▾ Case = Test 2
64 Fail message = Incorrect output!
65 input = 1e-13
66 0.4e-13
67 1e-22
68 output =      t:          x:          y:
69 [[ 0.00000000e+00 -6.00000000e-13  4.00000000e-14]
70 [ 1.00000000e-22 -5.98000151e-13  4.0000101e-14]
71 [ 2.00000000e-22 -5.96000454e-13  4.0000303e-14]
72 ...
73 [ 6.36000000e-20  5.94202694e-13  1.80659004e-13]
74 [ 6.37000000e-20  5.96151096e-13  1.81116754e-13]
75 [ 6.38000000e-20  5.98099632e-13  1.81574545e-13]]
76 Steps: 639
77
78 ▾ Case = Test 3
79 Fail message = Incorrect output!
80 input = 1e-13
81 2e-13
82 1e-20
83 output =      t:          x:          y:
84 [[ 0.00000000e+00 -6.00000000e-13  2.00000000e-13]
85 [ 1.00000000e-20 -4.01298823e-13  2.00432941e-13]
86 [ 2.00000000e-20 -2.05032477e-13  2.02081984e-13]
87 [ 3.00000000e-20 -1.34723547e-14  2.08369526e-13]
88 [ 4.00000000e-20  1.77277345e-13  2.27191442e-13]
89 [ 5.00000000e-20  3.72083864e-13  2.51212414e-13]
90 [ 6.00000000e-20  5.69142246e-13  2.76753731e-13]]
91 Steps: 71

93 ▾ Case = Test 4
94 Fail message = Incorrect output!
95 input = 1e-15
96 5e-15
97 1e-22
98 output =      t:          x:          y:
99 [[ 0.00000000e+00 -6.00000000e-13  5.00000000e-15]
100 [ 1.00000000e-22 -5.98000152e-13  5.00000127e-15]
101 [ 2.00000000e-22 -5.96000457e-13  5.00000382e-15]
102 ...
103 [ 6.36000000e-20 -4.64609823e-13  3.96653071e-13]
104 [ 6.37000000e-20 -4.66120832e-13  3.97964593e-13]
105 [ 6.38000000e-20 -4.67631951e-13  3.99276211e-13]]
106 Steps: 639
107
108 ▾ Case = Test 5
109 Fail message = Incorrect output!
110 input = 2e-14
111 2e-14
112 1e-24
113 output =      t:          x:          y:
114 [[ 0.00000000e+00 -6.00000000e-13  2.00000000e-14]
115 [ 1.00000000e-24 -5.99980000e-13  2.00000000e-14]
116 [ 2.00000000e-24 -5.99960000e-13  2.00000000e-14]
117 ...
118 [ 5.35750000e-20  1.43528563e-13  3.99958046e-13]
119 [ 5.35760000e-20  1.43536133e-13  3.99976354e-13]
120 [ 5.35770000e-20  1.43543703e-13  3.99994661e-13]]
121 Steps: 535781

```

Зураг 9. Жишээ VPL-дасгалын test case-үүд.

Хэрэв дасгалын бүтэн оноог 100 гэвэл, таван test case тус бүр 20 онооны байна. Жишээ нь оюутны бичсэн програм (**rutherford.py** файл)

гурван test case-ийг даваад үлдсэн хоёр дээр тэнцээгүй гэе. Тэгвэл 60 оноо авна гэсэн үг (Зураг 10).

The screenshot displays a Moodle VPL environment. On the left, a code editor shows a Python script named 'rutherford.py'. The script simulates Rutherford scattering using the Euler method. It defines physical constants like elementary charge (e), atomic mass unit (amu), and Coulomb constant (k). It sets simulation parameters such as initial time (t), atom mass (M\_atom), and charge (Q\_atom). The script takes user input for atom radius (R\_atom), impact parameter (b), and time step (dt). It then calculates the initial position and velocity of an alpha particle and simulates its path through a target region.

On the right, the 'Proposed grade' is 60 / 100. The 'Comments' section shows 'Failed tests' for Test 3 and Test 5, both with the error 'Incorrect output!'. A 'Summary of tests' box indicates that 5 tests were run and 3 were passed.

Зураг 10. Оюутны програмыг шалгаж, автоматаар үнэлж байгаа нь.

VPL-д одоогийн байдлаар, оюутан тухайн нэг дасгалыг 100 хувь тэнцүүлэх хүртлээ хязгааргүй олон удаа давтан гүйцэтгэж болно. [3]-т дурдсанаар, 30 орчим VPL-дасгалыг ажилласан мэдээлэл технологийн мэргэжлийн оюутнуудын гүйцэтгэлийн статистикийг харвал, 85 хувь нь 10-аас хэтрэхгүй удаа давтан ажиллаж, үлдсэн 15 хувь нь түүнээс олон удаа давтсан байна. Хамгийн ихдээ 51 удаа давтсан гэжээ. Үүнийг хязгаарлах нэг арга нь давталт болгонд торгуулийн оноо харгалзуулах юм. Түүнчлэн, шаардлагатай гэж үзвэл оюутны гүйцэтгэлийг test case-гэй тулгахаасаа өмнө эсвэл дараа нь, бусдаас

хуулсан эсэхийг нь бас шалгах боломжтой [3, 4].

### III. ДҮГНЭЛТ

Энэхүү өгүүлэл нь, Moodle VPL дээр “Тооцон бодох физик” хичээлийн дасгалыг хэрхэн үүсгэж бэлдэх техник ажиллагааны талаар авч үзсэн урьдчилсан (preliminary) судалгаа юм. Тооцон бодох физикийн хичээлд Moodle VPL хэрэгсэл ашигласнаар:

1) багш VPL-дасгалуудыг бэлдэхэд их цаг – хөдөлмөр зарцуулах ч, бэлдэж дууссаны дараа сургалтын үйл ажиллагаандаа хэрэглэхэд цаг - хөдөлмөр хэмнэнэ;

- 2) оюутнууд хичээлийн үеэр, хичээлийн дараа интернет байгаа ямар ч газраас дасгалуудыг ажиллах боломжтой;
- 3) ингэснээр оюутны сурч танин мэдэх дур сонирхлыг өдөөх, улмаар програм бичих дадлыг түргэн хугацаанд тэтгэн сайжруулах;
- 4) онлайнаар програм бичих шалгалт авахад нэн тохиромжтой;
- 5) дасгалын гүйцэтгэл ба үнэлгээг автоматаар хийснээр хүний (багшийн) субъектив нөлөөг арилгах боломжтой

гэх мэт давуу тал, шинэлэг нөхцөл байдлыг бий болгоно гэж үзэж байна. Нөгөө талаас,

- 1) багш ч, оюутан ч гэнэт ийм автомат үнэлгээний системд ажиллаж зүгшрэхэд тодорхой хугацаа шаардагдана;
- 2) багш, оюутны хоорондын амьд харилцааг багасгах эрсдэлтэй байж магадгүй;
- 3) VPL-дасгалыг давтан ажиллах тоог хязгаартай болгох шаардлагатай

болов уу. Гэхдээ үүнийг бодитоор үнэлэхийн тулд хангалттай олон тооны дасгал бэлдэх, тэдгээрийг хамгийн багадаа нэг бүтэн хичээлийн улирлын туршид сургалтад ашиглан харгалзах статистик өгөгдлийг цуглуулж, үр дүнгийн шинжилгээ хийх шаардлагатай. Тухайлбал:

- “Тооцон бодох физик” хичээл дээр уг хэрэгслийг бодитоор (дор хаяж бүтэн улирлын хугацаанд) ашиглан туршиж үзэх;
- Байгалийн ухааны салбарын оюутанд алгоритм ба програмчлалын үндсийг заах зорилготой “Шинжлэх ухааны програмчлалын үндэс” (PHYS215) хичээлд VPL хэрэгслийг ашиглан турших;
- “Загварчлал ба симуляци” (PHYS472) хичээлд ашиглан туршиж үзэх;
- VPL-дасгалын санг өргөжүүлэн баяжуулах (хэмжилтийн үр дүнгийн боловсруулалт, уламжлалт дифференциал тэгшитгэл, шугаман тэгшитгэлийн систем гэх мэт сэдвээр);
- VPL хэрэгсэлд суурилсан дасгалтай оюутан хэрхэн ажиллаж буй талаар дидактикийн судалгаа үнэлгээ хийх

гэх мэт. Эдгээр нь манай судалгааны ажлын дараагийн чухал үе шатууд байх бөгөөд тусдаа тайлан болох юм.

#### АШИГЛАСАН МАТЕРИАЛ

- [1] Moodle documentation, <https://moodle.org>
- [2] Virtual Programming Lab, <https://vpl.dis.ulpgc.es/index.php/en/>
- [3] Dominique Thiébaud, Automatic Evaluation of Computer Programs Using

Moodle’s Virtual Programming Lab (VPL) Plug-In, Journal of computer science in colleges, Vol. 30, No. 6, 146-151, 2015, doi: 10.5555/2753024.2753053

- [4] Vinicius F. C. Ramos et al., Student and Lecturer Perceptions of Usability of the Virtual Programming Lab Module for Moodle, Informatics in Education, Vol. 20, No. 2, 297–315, 2021, doi: 10.15388/infedu.2021.14
- [5] José Marílio Oliveira Cardoso, Integration Of Virtual Programming Lab In A Process Of Teaching Programming Eduscrum Based, PhD Thesis, 2020, <http://hdl.handle.net/10347/26051>
- [6] Azizah Zakiah et al., An Innovative Learning Approach for Enhancing Basic Programming Skills Through Virtual Lab, TEM Journal, Vol. 14, No. 2, 1629-1641, 2025, doi: 10.18421/TEM142-60
- [7] Marcus Messer et al., Automated Grading and Feed-back Tools for Programming Education: A Systematic Review, ACM Transactions on Computing Education, Vol. 24, No. 1, 1-43, 2024, doi: 10.1145/3636515
- [8] Aldo von Wangenheim et al., Developing Programming Courses with Moodle and VPL, The Teacher’s Guide to the Virtual Programming Lab, 1<sup>st</sup> Ed., Bookess, 2015.
- [9] Paulo Henrique Pisani, Andr’e C. P. L. F. de Carvalho, Applying the Virtual Programming Lab with Python, ICMC TECHNICAL REPORT, 2017, [https://web.icmc.usp.br/SCATUSU/RT/Relatorios\\_Tecnicos/RT%20419.pdf](https://web.icmc.usp.br/SCATUSU/RT/Relatorios_Tecnicos/RT%20419.pdf)
- [10] PhysMoodle – Сургалт удирдах систем, <http://124.158.127.55/moodle/>
- [11] Marvin L. De Jong, Introduction to Computational Physics, Addison-Wesley, 1991