



YECHIMNI AVTO-TESTLOVCHI TIZIMLARDAGI ASOSIY TERMINLAR. REYTINGLASH JARAYONING MATEMATIK MODELI.

Axmedova Nilufar Farxodovna

(Muxammad al-Xorazmiy nomidagi TATU Qarshi filiali "Axborot texnologiyalarining dasturiy ta'minoti" kafedراسi o`qituvchi - stajyor)

Samandarova Shaxnoza Mirjamilovna

(Muxammad al-Xorazmiy nomidagi TATU Qarshi filiali "Axborot texnologiyalarining dasturiy ta'minoti" kafedراسi o`qituvchi - stajyor)

Mamatqulova Dildora Farxodovna

(Muxammad al-Xorazmiy nomidagi TATU Qarshi filiali KI 12-19(sirtqi) talabasi)

Annotation: *In the article compilation in auto-test systems is the solution. Building a mathematical model of the rating process. Information on how to work with test files is provided.*

Annotatsiya: *Ushbu maqolada yechimini avto-test tizimlarda kompilyatsiya qilish. Reytinglash jarayoning matematik modelini qurish . Test fayllari bilan ishlash kabi ma'lumotlar keltirilgan.*

Аннотация: *Компиляция в системах автотестирования – решение в этой статье. Построение математической модели рейтингового процесса. Приведена информация о том, как работать с тестовыми файлами.*

Key words: *g++, Mingw MVS, Gcc, python -o mypython.py, Checker.*

Kalit so`zlar: *g++, Mingw MVS, Gcc, python -o mypython.py, Checker.*

Ключевые слова: *++, Mingw MVS, Gcc, python -o mypython.py, Checker.*

Yechimni avto-testlovchi tizimlarda qaysidir terminlarni joriy etish uchun operatsion tizim va unda o`rnatilgan barcha kompilyatorlarni yoki interpretatorlarning ishlash prinsipi bilan tanishib chiqamiz. Kompilyatorlarning operatsion tizim bilan bog`liqliligini, kompilyatsiya va interpretatsiyaning amalga oshirish jarayonlarga qarab ishlab chiqarayotgan tizimimiz qaysi turdagi javoblarni taqdim etishi mumkinligini aniqlashtiramiz. Albatta barcha jarayonlarni `rganib chiqqandan keyin terminlarni xorijiy va mahalliy shunga o`xshagan tizimlardagi kabi terminlar bilan standartlashiramiz. Eng avallo kompilyator va interpretatorlarni asosiy prinsiplari bilan tanishamiz.

Ma'lumkin kompilyator yoki interpretatorlar kiritilgan inson tiliga yaqin leksimalarni mashina tiliga o`tkazib beradilar. Albatta bu jarayonni shunchaki 4-5 ta so`z orqali ifodalash mumkin emas. Kompilyatorlar ham interpretatorlar ham maxsus bajaruvchi fayl ko`rinishda bo`ladi. Masalan C/C++ kompilyatorlariga g++, Mingw MVS, Gcc kiradi. Ularning bajaruvchi fayli bajaruvchi satrdan bir



necha parametr va fayl nomini olib parametrlarga mos holda natija qaytaradi. Python kabi dasturlash tilining interpretatori ham xuddi shu tarzda ishlaydi. Masalan python da mypython.py nomli faylimizni bajarmoqchi bo'lsak, u holda python interpretatoriga o'tib python -o mypython.py buyrug'inin berishimiz kerak. Natijada python interpretatori ishga tushin bizning yozgan kodimizni bajaradi. Boshqa statik va dinamik tiplashgan dasturlash tillari ham xuddi shu kabi tarzda ishlaydi. Yana buyrug satri orqali fayl nomini olib natijani chiqaradi.

Kompilyatorlar va interpretatorlarni oddiy IDE lardan ham o'rganish mumkin. Bunda biz IDE ga kod yozamiz yozilgan kodimiz IDE tarafidan kompilyator yoki interpretator tomon yo'naltiradi. Natijani esa IDE bizga taqdim etadi. Umumiy ma'noda kelgusida ishlab chiqarmoqchi bo'ladigan web platformamizham xuddi shu tarzda ishlashi kerak ha albatta bizning holda online va real vaqt tizimida ishlashi kerak. Xo'sh kompilyatorlar yoki interpretatorlar yozgan kodimizni qayta ishlab qanday turdagi natijalar qaytarishi mumkinligini o'rganamiz.

Agar yozilgan kodimizda sintaktik xatolik bo'lsa bajaruvchi amaliy dastur (kompilyator yoki interpretator nazarda tutilmoqda) shu xatolik joylashgan qatorni, xatolik turini va ba'zi hollarda yechuvchi kodni (masalan mvs kompilyatori) qaytaradi. Barcha natijalar stdout yane standart chiquvchi faylga yoziladi. Agar kodimizda sintaktik xatolik mavjud bo'lmasa u holda stdout ga qandaydir natijani chop etadi. Yozilgan yechimimizda xech qanday xatolik bo'lmagan taqdirda mazmunan va mantiqiy xatolik bo'lishi mumkin va bularni bartaraf etish uchun qo'yidagi tushunchalarni kiritamiz.

Checker-yoki jyuriylar quyilgan testlarga mos ravishdagi yechimning javoblarini tekshiruvchi qism. Quyilgan masala yoki muammoga mos ravishda tog'ri keladigan yechimlar to'plamini tekshirish asosan ikki xil usulda bo'ladi.

ACM usuli

IO usuli

ACM usuli hozirgi zamondagi yechimni avto-testlovchi tizimlarda juda keng qo'llaniladi. Bu telshiruv usulida qo'yilgan masalaga mos ravishda yechim tuziladi. Yechim masala muallifi yoki juriylar tarafidan tuziladi va foydalanuvchi junatgan yechim bilan paralel ravishda bajariladi va uning to'g'ri yoki noto'g'ri ekanligini real vaqtda tekshiriladi. Bunday tarzdagi usul odatda katta miqyosdagi musobaqalarda qo'llaniladi. Masalan ulardan biri bu ACM/ICPC xalqaro dasturlash bo'yicha olimpiada. Bu Xalqaro olimpiada ICPC LIVE tarzda o'tkaziladi va ACM-usulda bo'lib o'tadi. Bu usul hozirgi paytda ko'pgina IT gigantlar tarafidan qo'llab quvatlanadi. ACM usuliga asoslangan texnologiya yilma yil yangilanib musobaqalarda qo'llanib kelmoqda.



IO-kiritish va chiqarishga mo'ljallangan turdagi tekshirish usulida qo'yilgan masala yoki muammo fayllar asosida tekshiriladi. Bunda foydalanuvchi kiritish va chop etish oqimini jyuriylar tomonidan qo'yilgan talabga mos ravishda boshqa bir faylga yunaltirishi mumkn. (Masalan input.txt output.txt). Lekin bu majburiy emas hozirda ko'pgina operatison tizimlar standart oqimlar (stdin, stdout) ni bash yoki bat fayllar yordamida boshqa faylga yo'naltirishlari mumkin. Bu usulda muammo yoki masalaning yechimlari oldindan qaysidir faylga joylashtiriladi. Bu fayllar test fayllar deyiladi. Test fayllar bir nechta bo'lishi mumkin. Foydalanuvchi yechimidan hosil bo'lgan natija aynan shu fayl orqali tekshiriladi. Fayl tekshirish alohida yozilgan utilita orqali amalga oshiriladi. Agar foydalanuvchi yechimi fayldagi natiha bilan belgima-belgi mos tushsa u holda yechim to'g'ri deb qabul qilinadi, aks holda aynan qaysi test fayldan yechim o'tilmagani tizim tarafidan ma'lum qilinadi. Bu usul kichik musobaqalarni o'tkazishda va o'rganish maqsadida kichik va aniq yechimga ega bo'lgan masalalarni yechishda qo'llaniladi.

IO-formatdagi musobaqa ACM -formatdagiga nisbatan ko'pgina kamchiliklarga ega.

Yechimi faqatgina bir yo'nalishda bo'lgan muammo yoki masalalarni yechishda qaratiladi.

Faqatgina fayl yoki standart oqimdagi ma'lumot orqali tekshirish imkoniyati mavjud.

Interaktiv masalalarni yechishda qo'lashni iloji yo'q.

Natijalar belgilar asosida tekshiriladi. Yane foydalanuvchi oddiy bitta simvol masalan bo'sh joy qolib ketsa ham yechim xato hisoblanadi.

Muamoning yok masalaning muallifi oldindan test fayllar tuzishi kerak bo'ladi.

Foydalanuvchi yuborgan yechim qaysi uslubda teksirilmasin, tizim foydalanuvchiga yechimlar asosida javob qaytarishi kerak bo'ladi. Foydalanuvchi muamoning yechimini jo'natgandan so'ng tizim foydaluvchiga qandaydir javob qaytarishi kerak. Bu uchun albatta ISO xalqaro standartlarga mos ravishda javob terminlarini aniqlashimiz kerak.

Verdikt-bu foydalanuvchi yuborgan yechimga mos bo'lgan javob. Demak asosiy verdiktlarni foydalanuvchi va jyuriylar yechimidan qo'yidagilarni aniqlaymiz.

1.2.1-jadval. Tizimdagi asosiy verdiktlar

Verdikt nomlanishi	Olingan verdikt ma'nosi
--------------------	-------------------------



Accepted	Bu verdikt foydalanuvchi yechimi test fayllar yoki jyuriylar yechimi bilan mos tushganda qaytariladi
Time Limit Exception (TLE)	Bu verdikt foydalanuvchi yechmi aniq bir masala yoki muammoga qo'yilgan vaqt chegarasidan o'tmaganligini bildiradi.
Memory Limit Exception (MLE)	Bu verdikt foydalanuvchi yechimi aniq bir masala yoki muammoga qo'yilgan xotira hajmi chegarasidan o'tmaganligini bildiradi.
Wrong Answer (Test Num)	Bu verdikt foydalanuvchi yechimi jyuriylar yechimining qaysidir test faylidan o'tmaganligini bildiradi
Partial accepted	Bu verdikt foydalanuvchi yechimi asosiy yechim bilan qisman mos ekanligini bildiradi.
Compilation Error Exception CER	Bu verdikt foydalanuvchi yuborgan yechim sintaktik xatolikka ega ekanligini ko'rsatadi.

Har bir test fayllar jyuriylar tomonidan qo'yilgan umumiy ballga ega. Bunga albatta natijaviy ball (Score) sifatida barcha testlarni soniga umumiy ballni ko'paytmasi anqilanadi.

$$\text{Score} = \text{CmBall} * \text{CntTest}$$

Bu yerda Score-natijaviy ball, CmBall-bir dona test faylga beriladigan ball, CntTest- esa test fayllar soni. Bu faqatgina masalalar yoki qo'yilgan muammolarni



yechadiganda beriladigan ballar soni hisoblanadi. Bu ballar tizimdagi umumiy ballar va reytinglash tizimiga ta'sir qilmaydi. Tizimda reytinglash jarayoni faqatgina musobaqalar tashkillashtirish vaqtida amalga oshiriladi. Albatta bu musobaqani tashkilashtiruvchilar tarafidan amalga oshiriladi.

Yechimni avto-testlash dagi asosiy terminlar va reytinglash tizimini muayyan miqdorda xorijiy va mahalliy dagi tizimlardagidek moslashtirdek. Albatta bu kelajakda tuzilayotgan web platformamizni life-cycle yane hayot siklni uzaytirishga yordam beradi.

Xulosa : Tizimdagi asosiy verdiktlarni aniqlab olamiz va foydalanuvchilarni rag'batlantirish uchun reytinglash tizimini matematik modelini tuzib olamiz. Yuqorida ta'kidlaganimizdek tizim foydalanuvchilar yechimini test fayllar orqali aniqlaydi. Har bir test fayllar jyuriylar tomonidan qo'yilgan umumiy ballga ega. Bunga albatta natijaviy ball (Score) sifatida barcha testlarni soniga umumiy ballni ko'paytmasi anqilanadi.

$$\text{Score} = \text{CmBall} * \text{CntTest}$$

Bu kelajakda tuzilayotgan web platformamizni life-cycle yane hayot siklni uzaytirishga yordam beradi.

FOYDALANILGAN ADABIYOTLAR:

Sh.M.Samandarova, E.E.Jabborov "THE PRIORITY OF DEVELOPING THE INTELLECTUAL POTENTIAL OF STUDENTS IN THE CONDITIONS OF INFORMATIZATION OF EDUCATION" "International scientific research conference" belarus, international scientific-online conference. MINSK-2023

.M.Самандарова, А.З.Каримова "БИОСИГНАЛЛАРГА ЛАГРАНЖ ИНТЕРПОЛЯЦИЯЛАШ УСУЛИ АСОСИДА ИШЛОВ БЕРИШ АЛГОРИТМИ" MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI QARSHI FILIALI Sun'iy intellekt va raqamli texnologiyalarning jamiyatdagi o'rni mavzusida respublika ilmiy-amaliy anjumani ma'ruzalar to'plami 2023 yil 4-5-may