

# Први домаћи задатак из Генетских алгоритама

## 1. Задатак

Имплементирати **дати** генетски алгоритам за решавање **датог** проблема у програмском језику по **избору**. Конкретан алгоритам и проблем узимате на основу ознаке теме у дељеном документу са поенима (<https://rafcs.xyz/ga2023> - за приступ морате бити пријављени преко @raf.rs налога).

### 1.1. Документација

Саставити кратку документацију за инсталацију и покретање задатка (ради понављања експеримента – тестирања). Документација треба да садржи:

- Језик у коме је пројекат написан
- Коришћене екстерне библиотеке (ако постоје)
- Начин конфигурисања програма
- Како се код може компајлирати и покренути

## 2. Имплементација

Да бисте освојили све поене за имплементацију, морате испунити одређене услове.

### 2.1. Подесивост

Програм мора да буде потпуно подесив без икакве промене у коду, односно без потребе да се поново компајлира. То значи да је сама имплементација фиксирана за:

- Врсту алгоритма који сте имплементирали
- Конкретан проблем који решавате

Имплементација није везана за:

- Број узастопних поновних покретања програма на истом проблему
- Величину популације
- Број јединки за парење
- Број јединки за мутацију
- Број јединки које се избацују или праг одсецања за селекцију
- Нумеричке параметре функције прилагођености/трошка (ако постоје)
- Random seed за генератор случајних бројева
- Друге сличне нумеричке параметре
- Путању до излазне текстуалне датотеке

Сви параметри за које имплементација није везана (дакле све горе набројано) треба да буду лако подесиви. Препоручује се коришћење текстуалног конфигурационог фајла који **није део кода**, а који се подразумевано учитава. За све параметре у коду усвојити подразумеване вредности у случају да се неки од тих параметара не проследи и то обавезно документовати (које су вредности у питању). Омогућити прослеђивање путање до конфигурационог фајла преко командне линије (ако није прослеђен, гађа се

подразумевани, а ако ни њега нема, онда се учитавају подразумеване вредности из кода). Нека програм буде отпоран на грешке конфигурације.

## 2.2. Испис резултата

У току извршавања програм у излазну текстуалну датотеку на путањи из конфигурације исписује за сваку генерацију прилагођеност најбољег решења и просечну прилагођеност популације.

- Најбоље решење у генерацији је решење које има најбољу прилагођеност или најмањи трошак
- Просечна прилагођеност популације у једној генерацији је средња вредност прилагођености (или трошкова) свих чланова те популације у тој генерацији

На крају извршавања програма исписује се састав најбољег хромозома. У случају бинарног генетског алгоритма, поред састава хромозома (нуле и јединице) исписују се и фенотипске вредности гена – декодовани хромозом. У случају континуалног алгоритма урадити исто тако ако фенотип није директно пресликан на генотип.

Ако путања датотеке није конфигурисана или датотека не може да се отвори, програм исписује све те резултате на стандардни излаз.

## 3. Презентација

Да бисте освојили све поене за презентацију, морате испунити одређене услове

### 3.1. Садржај презентације

На првом слајду обавезно написати: име, презиме и број индекса (у форми СМЕР-БРОЈ/ГОДИНА).

Надаље написати поставку конкретног задатка (тј. који проблем је решаван и којим алгоритмом). Затим следи опис вашег решења, у коме треба навести све што је релевантно (почевши од одабраног језика, затим каква функција прилагођености или функција трошка је коришћена, како су хромозоми кодовани ако постоји кодовање, који су нумерички параметри искоришћени за оно што није фиксирано поставком задатка и како сте до њих дошли). Напишите опис комплетног експеримента на основу кога сте закључили да су ваши параметри најадекватнији за конкретан проблем.

Изложити резултате добијене тестирањем са тим параметрима, при чему обавезно треба направити прекид после 150 генерација ако алгоритам не конвергира раније. Приказати резултате три експеримента са различитим величинама популације: 10 хромозома, 50 хромозома и 90 хромозома.

Сваки појединачни експеримент садржи три покретања са истим подешавањима и са истим бројем хромозома. Резултате приказати графички на следећи начин:

- График прилагођености најбољег решења у популацији у односу на редни број генерације
- График просечне прилагођености популације у односу на редни број генерације

Све графике истог типа из истог експеримента ставити на исти оквир, исцртане различитим бојама.

Требало би да имате укупно 6 оквира, по два за сваки експеримент, а на сваком оквиру по три линије, где свака линија представља једно покретање алгоритма са одговарајућом конфигурацијом. Погледајте пример графика на крају овог документа.

### 3.2. Одбрана

Одбрана је усмена, а састоји се из излагања презентације и образложења добијених резултата. Биће потребно и да се кратко прође кроз код и по потреби нешто детаљније објасни или измени на лицу места. Било какво неразумевање сопственог кода неће бити толерисано (одузимају се поени). Образложења

типа „радио сам пре две недеље, не сећам се” и слично не долазе у обзир. Пре почетка рада водите рачуна о томе да сте добро разумели тему и захтеве из спецификације и поставите питања уколико нешто од наведеног није јасно.

## 4. Предаја рада

Рад треба да садржи **имплементирани алгоритам, конфигурациони фајл, презентацију решења и документацију**, у једној архиви. Архиву послати са **свог студентског мејла** на [mtomic@raf.rs](mailto:mtomic@raf.rs) (у наслову поруке напишите да је први домаћи из Генетских алгоритама). Дозвољени језици за имплементацију су: C, C++, Java, Python, JavaScript, Octave.

Меки рок за предају задатка је **13. април 2023. у 12.00 (подне)** – четвртак пред почетак треће колоквијумске недеље. Предаја до овог рока може донети максимално 100% поена које носи овај задатак, односно 15 поена. Сваки минут закашњења умањује максимални број поена за 0,0125 (задаци предати после 14. априла у 8.00 вреде 0 поена). **Исправна имплементација са свим наведеним ставкама** носи 70% поена, а **исправна презентација (са одбраном)** 30% поена. **У случају да одбрана не буде успешна, поени са имплементације се не рачунају** (тј. задатак носи 0 поена).

Сва своја питања у вези са задатком можете слати мејлом. Очекивано време за одговор је 24 сата од тренутка када сте послали мејл, осим ако је викенд (тада се продужује на први следећи радни дан). Очекујте да од ваших питања формирам документ са постављаним питањима и проверите га пре него што поставите питање, за случај да је на ваше питање већ одговорено.

## 5. Додељене теме и завршне напомене

За селекцију пред укрштање можете користити било коју осим турнирске.

Ознаке тема (колона ЗАДАТАК 1 у документу подељена на четири колоне) састоје се из 4 слова:

Прво слово означава тип генетског алгоритма

Друго слово означава методу укрштања

Треће слово означава методу мутације

Четврто слово означава проблем који се решава

У наставку су дате табеле ознака.

### 5.1. Типови генетског алгоритма

Б – бинарни генетски алгоритам

К – континуални генетски алгоритам

### 5.2. Методе укрштања

Ј – једнотачкасто укрштање на случајној тачки укрштања

Д – двотачкасто укрштање на случајним тачкама укрштања

У – униформно укрштање

М – метода једноставног мешања за континуални генетски алгоритам (Редклиф), са случајно изабраним параметром

A – BLX- $\alpha$  укрштање за континуални генетски алгоритам

X – хеуристичко укрштање за континуални генетски алгоритам

### 5.3. Методе мутације

T – тачкаста мутација (флиповање произвољних битова) за бинарни генетски алгоритам

I – инверзија за бинарни генетски алгоритам

Z – тачкаста униформна мутација за континуални генетски алгоритам (бира се случајна вредност из домена гена по униформној расподели)

N – тачкаста нормална мутација за континуални генетски алгоритам (додаје се случајна вредност на ген из нормалне расподеле)

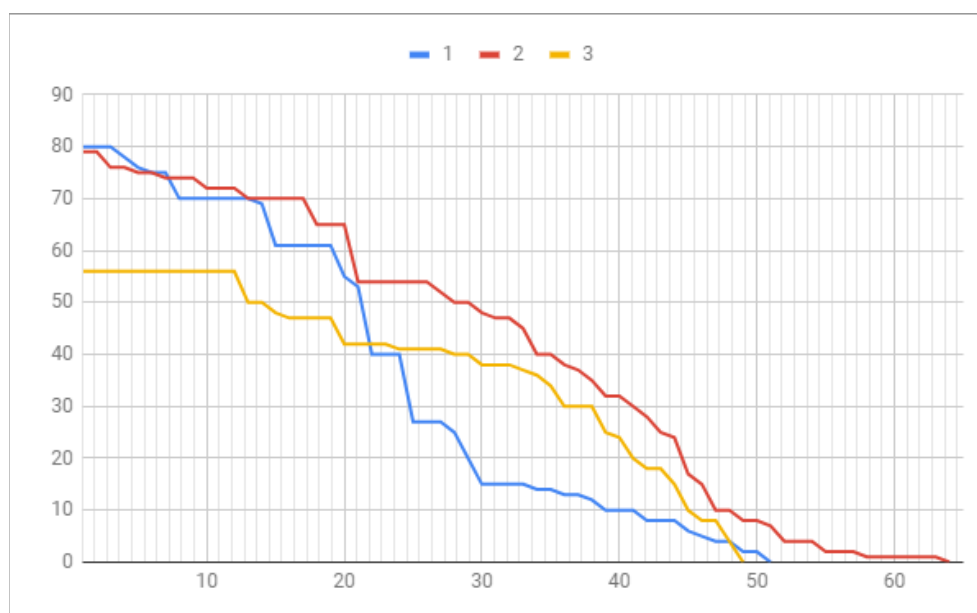
### 5.4. Проблеми

**М, Л, Г, К, Б:** Пет конфигурација неуронских мрежа које се тренирају на одређеном сету података. Улазни параметри за покретање мреже су вредности тежинских коефицијената грана у мрежи који су дефинисани у наредној секцији упутства. Излазна вредност приликом покретања симулације биће вредност функције грешке. На вама је да минимизујете грешку подешавањем конфигурационих параметара.

### 5.5. Графички приказ резултата

Пример графика најбољег решења у популацији за један експеримент (три покретања) у односу на број генерација приказан је на доњој слици. За овај график послужио је MS Excel, а на вама је како ћете своје графике изградити (при чему за сваки график узимате одговарајуће податке из излазне текстуалне датотеке). Можете и свој програм направити тако да вам аутоматски при извршавању генерише графике (то је најпрактичније решење).

На x-оси је редни број генерације, на y-оси је вредност функције за најбоље решење у одређеном покретању у тренутној генерацији.



## 6. Проблеми

Сваки задатак је неуронска мрежа коју треба компајлирати из изворног кода и онда повезати са својим решењем како би се извршило тренирање. Мрежу користите за израчунавање функције трошка, коју у овом случају треба минимизовати. Тежински коефицијенти неуронске мреже задају се као аргументи командне линије. Резултат провере мреже (тј. вредност функције трошка за појединачни случај) добија се на стандардном излазу и одатле га треба прочитати. У принципу не постоји ограничење за тежинске коефицијенте, можете узети да су вредности од -3 до 3, са прецизношћу на произвољан број децимала који вам даје повољне резултате (најмање једно место иза децималне тачке, али може и више). Није позната минимална изводљива вредност функције трошка (у идеалном случају је 0).

### Б

Фајл b.c, тренира се са укупно 27 тежинских коефицијената који се задају преко командне линије (реални бројеви раздвојени размаком).

### Г

Фајл g.c, тренира се са укупно 33 тежинска коефицијента који се задају преко командне линије (реални бројеви раздвојени размаком).

### К

Фајл k.c, тренира се са укупно 33 тежинска коефицијента који се задају преко командне линије (реални бројеви раздвојени размаком).

### Л

Фајл l.c, тренира се са укупно 32 тежинска коефицијента који се задају преко командне линије (реални бројеви раздвојени размаком).

### М

Фајл m.c, тренира се са укупно 32 тежинска коефицијента који се задају преко командне линије (реални бројеви раздвојени размаком).