

Стефан С. Павић¹
Универзитет у Крагујевцу
Филолошко-уметнички факултет
ORCID 0009-0000-4086-4207

БИСЕР И АВАТАР – БЛИСКО ЧИТАЊЕ КОДНЕ ПОЕЗИЈЕ²

Циљ овог рада јесте да анализира шта заправо чини кодну поезију, представи које су то компликације и проблеми приликом самог дефинисања појма кодне поезије, као и да створи савремену категоризацију песама писаних програмским језицима, ослањајући се примарно на теоријски текст ауторке Мари-Лор Рајан *Аватари приче*. Рад ће такође настојати да осавремени и употпуни наведена теоријска разматрања имајући у виду новонастале програмске језике и нове форме кодне поезије. На послетку, користећи методу блиског читања на песми „Бисер” (*Black Perl*), рад ће покушати да укаже на то која је главна функција игре и мешања људског и програмског језика у оквиру једне песме и на који начин управо то мешање уноси додатне слојеве значења и може продубити емотивни утисак који песма оставља.

Кључне речи: кодна поезија, блиско читање, Мари Лор Рајан, програмски језик, металепис, код

Како би применили концепт блиског читања на било коју песму која припада кодној поезији неопходно је најпре установити шта је заправо кодна поезија. Већ на почетку, код саме дефиниције постоје одређени проблеми јер се ради о релативно новом и поприлично компликованом виду песништва који захтева по-дробно разумевање како програмских језика тако и конвенција традиционалне поезије. Како Мери-Лор Рајан (2009: 218) у књизи *Аватари приче* објашњава:

Понашање сваког дигиталног текста је регулисано невидљивим кодом програма. Присуство овог кода отвара могућност да се оде један ниво испод и да се игра оним што се може назвати 'дигиталним доњим вешом' текста. Овај гест је карактеристика која дефинише један покрет у књижевности нових медија познат као 'кодна поезија'³.

Дакле, могло би се рећи да је ова видљивост кода заправо главна одлика кодне поезије. Ауторка такође повлачи, на неки начин, паралелу између рачунара, програмских језика, па самим тим и кодне поезије и металепсе. Металепа је, како се наводи, „реторичка и наративна фигура описана још у раном седамнаестом веку” (Рајан 2009: 204). То би заправо представљало, барем у књижевности, различите нивое наратије као и границе између тих нивоа. Као објашњење за ово Мери-Лор Рајан наводи да, ако имамо промену наратора који ступа на сцену и објашњава шта га је ту довело, то спада у илокуциону границу, док за другу врсту, онтолошку, издваја Шехерезаду из *Хиљаду и једне ноћи* и објашњава како је ту границу заправо у оквиру самог наратива који ствара нови фикционални свет. Наиме, Шехерезада у оквиру једног наратива ствара нови наратив сваком

1 stefan.pavic@filum.kg.ac.rs

2 Истраживање спроведено у раду финансирано је Министарство науке, технолошког развоја и иновација Републике Србије (уговор о реализацији и финансирању научноистраживачког рада НИО у 2024. години бр. 451-03-66/2024-03/ 200198)

3 У оригиналном тексту ауторка користи реч *codework* која се користи као синоним за кодну поезију (на енглеском *code poetry*).

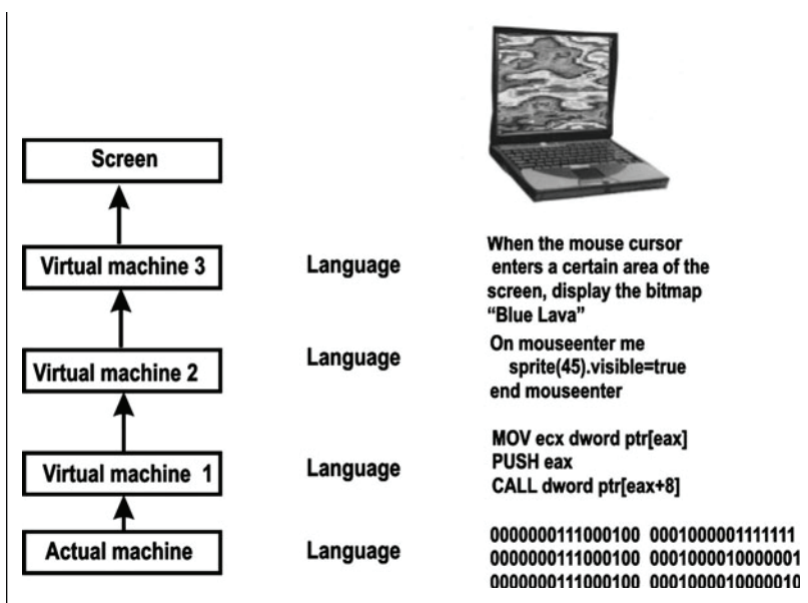
новом причом, а у оквиру једне од њених прича јавља се служавка која пак ствара свој наратив (Рајан 2009: 206). Када се тај наратив заврши прелази се на следећи ниво изнад тог наративног нивоа, у том смислу можемо рећи да се металепа у случају књижевног наратива „креће или захвата на горе” зато што се од најдубљег слоја креће према првом наративу који је све започео. Ту заправо говоримо и о вези овог концепта са етимолошким коренима саме речи, што и Мери-Лор Рајан (2009: 206) примећује: „Етимологија нам говори да се термин састоји из два Грчка корена: префикса *meta* ’оно што је изнад или обухвата’ и суфикса глагола *ламбанеин*, ’захватити.’” Паралела се може повући између тога и начина на који се нужно чита кодна поезија. Код кодне поезије креће се од површинског текста ка дубљим слојевима значења као што су разумевање команди, функција и слично. Важно је овде направити дистинкцију да се не ради још увек о било каквој интерпретацији песме већ само о њеном читању.

Оставља се питање да ли је заправо оваква кретња кроз текст заправо металепа по строгој дефиницији исте, јер се ипак не ради о оформљеним наративима и евидентно постоје разлике, али се такође и олако може увидети веза између ова два, нарочито ако се узме у обзир онтолошка металепа код које се не види јасна граница између слојева наратива. „Можемо упоредити реторичку⁴ металепсу са бенигим израштајем који не утиче на суседна ткива, а онтолошку металепсу са инвазивним израштајем који уништава околна ткива.” (Рајан 2009: 206–207)

Међутим, и даље је тешко дати једну општу и сведену дефиницију тога шта заправо све може представљати кодну поезију, као и у којој форми она може бити изведена, то јест написана. Металептички елемент, као и видљивост кода, несумњиво морају постојати али постоји и прегршт преступа, а, како нам сама ауторка говори, „књижевност цвета на преступима, поготово у савременом добу” (Рајан 2009: 206). У зависности од самог програмског језика који се користи, форма песме ће се значајно мењати, где притом не говоримо само о структури стихова или рими већ о још конкретнијим променама које се виде и визуелно, али које такође утичу и на начин на који се та песма чита, па на послетку и на наше разумевање исте, због тога што ће умногоме зависити од разумевања и упознатости са самим програмским језиком.

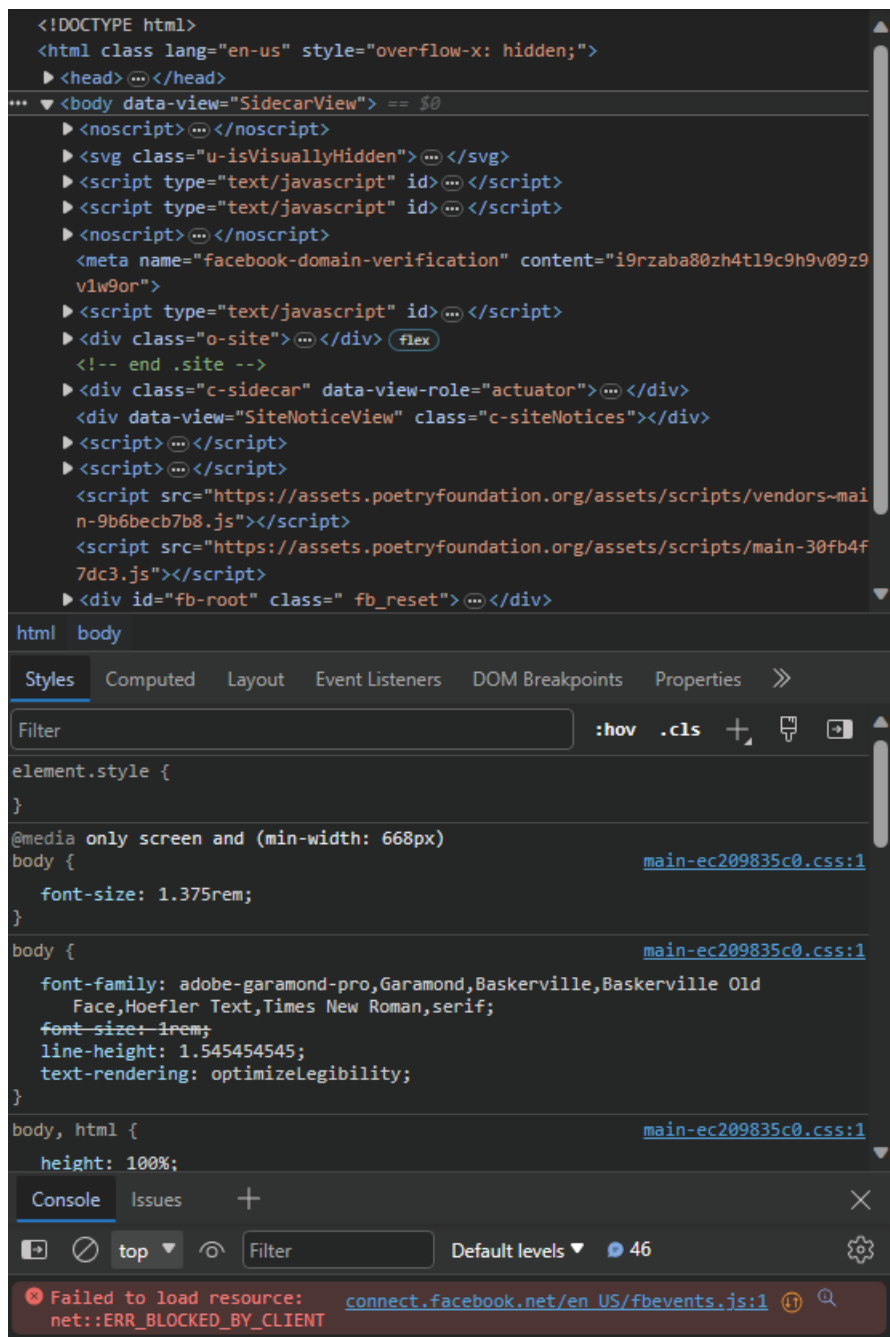
У својој књизи Мери-Лор Рајан даје илустрацију различитих нивоа језика на којима рачунари функционишу и то наводи као још један пример металепе. Исти се графикон може употребити и за илустрацију тога на који начин све разлике између програмских језика утичу на сам текст, дојам и разумљивост кодне поезије.

4 Под овим се мисли на врсту металепе код које су границе између различитих нивоа наратива јасне.



Графикон број 19. који Мери-Лор Рајан (2009 :217) користи као илустрацију металеписе у оквиру програмских језика и кодова

Такође, данас је могуће отворити било које дело које припада неком од конвенционалних праваца поезије у интернет браузеру и заправо видети код којим је то дело преточено на интернет страницу. Технички говорећи, тај видљиви код јесте песма написана неким програмским језиком (најчешће ХТМЛ језиком), међутим, не би било праведно тако нешто назвати кодном поезијом из више разлога. Првенствено јер не постоји игра између људског и програмског језика, затим се не ради о оригиналном делу, а сам код је дизајниран да буде невидљив, није фокус дела и форме већ само има функцију да нам прикаже одређени садржај на интернет страници.



Као пример можемо навести слику HTML кода Тенисонове песме “Crossing the Bar” преузето са <<https://www.poetryfoundation.org/poems/45321/crossing-the-bar>>.

Ово заправо представља ништа више до модернизације примера који Рајанова даје где говори о томе како се може видети скривени код из програма чији је била коаутор.

У неком тренутку у тексту, корисница истражује напуштени ранч тако што кликне на собе на карти. Али ако она стави миша преко скривеног места, на екрану ће се појавити код из Директора са којим је ова конкретна слика на екрану створена, као и псеудо превод асемблерског језика кода из Директора. Ово завиривање иза сцене показује како је текст створен, али не приказује код као уметничко дело. (Рајан 2009: 218)

Када се, имајући све ово у виду, нешто ближе одреди како код заправо треба да изгледа у кодној поезији како би створио уметничко дело, може се и доћи до одређене категоризације. Па на првом месту можемо ставити све оне песме код којих је код полуфункционалан или нефункционалан. Другим речима, када се код убаци у програмски језик, резултат тог кода не даје ништа, ништа се не дешава и не извршавају се команде јер сам код није довољно правилно написан већ представља хибрид између људског и програмског језика који нема функцију. Пример за тако нешто је песма „Црни Перл” или „Црни Бисер” (ради се о игри речи јер је *перл* програмски језик али звучи као енглеска реч за бисер), односно “Black Perl” у оригиналу, о којој ће бити више речи у даљем тексту.

```
BEFOREHAND: close door, each window & exit; wait until time.
    open spellbook, study, read (scan, select, tell us);
write it, print the hex while each watches,
    reverse its length, write again;
kill spiders, pop them, chop, split, kill them.
    unlink arms, shift, wait & listen (listening, wait),
sort the flock (then, warn the "goats" & kill the "sheep");
kill them, dump qualms, shift moralities,
values aside, each one;
    die sheep! die to reverse the system
    you accept (reject, respect);
next step,
    kill the next sacrifice, each sacrifice,
    wait, redo ritual until "all the spirits are pleased";
    do it ("as they say").
do it(*everyone***must***participate***in***forbidden**s*e*x*).
return last victim; package body;
    exit crypt (time, times & "half a time") & close it,
    select (quickly) & warn your next victim;
AFTERWORDS: tell nobody.
    wait, wait until time;
    wait until next year, next decade;
        sleep, sleep, die yourself,
        die at last
```

(Песма "Black Pearl", преузето са <<https://codedocs.org/what-is/black-perl>>)

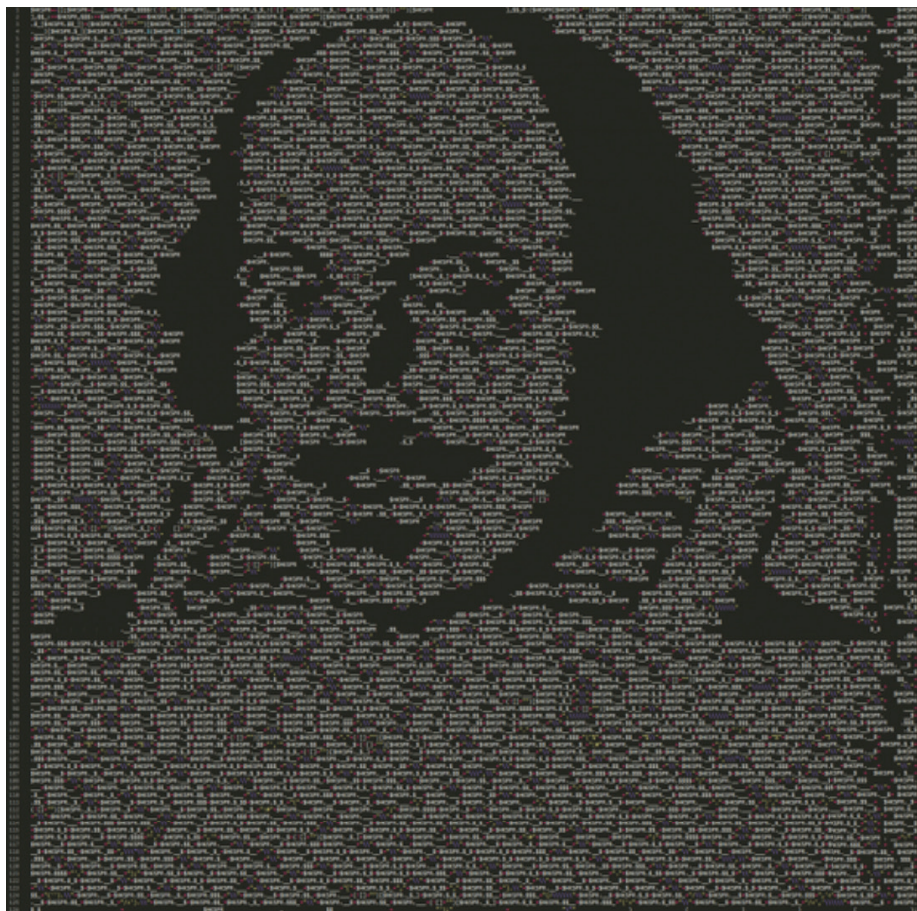
На другом месту би биле песме са функционалним кодом који је у потпуности написан по правилима програмског језика који се користи. Међутим, и ту постоје одређене компликације и несигурности. Постоје одређене ситуације где би се врло тешко могли рећи за неку песму да припада кодној поезији чак иако је написана у оквиру функционалног, правилног кода. На пример, могуће

је цео текст песме ставити као коментар или као већ одређени аутпут неког кода који нема везе за самом песмом или поезијом. У неким програмским језицима је такође могуће само користити команду одштампај и потом убацити цео текст песме на људском језику. У тим случајевима говоримо о правилном коду али не и о игри између људског и програмског језика и њиховом мешању, аналогном онтолошкој металепси. Иако коегзистирају у оквиру истог кода, у таквим случајевима не постоји интеракција између кода и људског језика. Дакле, у том смислу у овој категорији припададе би само песме које испуњавају и тај услов у оквиру једног правилно написаног кода. Пример за то може бити песма са називом „Вода” односно “Water”:

(преузето са <<https://code-poetry.com/water>>)

Трећа категорија би биле песме које се граниче са оним што се у теорији књижевности назива конкретна поезија. Ту се лингвистички елементи користе како би се од текста дословно изградила слика на папиру. Овакву поезију су на пример писали Џорџ Херберт, Роберт Херик, Луис Керол, Едвард Естлин Камингс и тако даље. Међутим овде се већ наилази на проблеме при тражењу примера овога у кодној поезији. Генерално говорећи, песама писаних у програмским језицима нема пуно јер се ради о, барем релативно, новом, експерименталном правцу који спаја књижевност са информационим технологијама, а заједнички пресек ове две науке није нарочито велики у смислу да није изнедрио велики

број аутора који би се бавили оваквом поезијом. Конкретно, ова врста поезије је још ређа и мање запажена, поготово када узмемо у обзир да практично и не постоје научни текстови о кодној поезији који се баве специфично оваквом врстом песама. Као један пример за конкретну кодну поезију, како је можемо назвати, могуће је узети управо претходни пример из друге групе јер се поред тога што поседује правилно написан код такође види да се заграде и тачке и зарези плаве боје врло свесно користе као репрезентација воде. Још један пример, више као илустрација тога шта је могуће, био би следећи:



(преузето са <<https://www.sourcecodepoetry.com/?pgid=jl4tl4ne-f6094dd4-ca93-47e8-a5ce-4596a55262aa>>, аутор је Денис Зик, међутим проблем је у томе што се у оригиналу заправо не види текст и није било могуће доћи до бољег примерка где би се текст заправо видео. Сходно томе ово служи само као илустрација шта је све могуће урадити кодним језиком у смислу конкретне поезије.)

Имајући у виду ове различите категорије кодне поезије, природно се намеће питање функције и првенствено разлога иза оваквог мешања песништва и програмирања. Наиме, да ли је у оваквим случајевим код само нека врста колажа, украса или заправо има и такорећи књижевну функцију да прошири значење песме, појача или промени одређена осећања која песма изазива код читалаца и

томе слично. Како би се дошло до овог одговора, неопходно је заправо бавити се анализом неке конкретне песме. Мери-Лор Рајан се благо дотиче ове проблематике када се позива на аутора Џона Кејлија који објашњава зашто, по његовом мишљењу код у песми мора имати функцију ван само естетске и украсне већ мора дати резултат, другим речима мора бити правилно написан и заправо урадити нешто када се покрене (Рајан 2009: 219–221). Доктор Јанез Стреховец (2003: 1–3) у тексту „Текст као петља / О дигиталној поезији” говори о томе како је заправо функција кода у дигиталној поезији да допринесе језичкој економичности, како би се што више рекло у што мање речи, као и да разбије линеарност читања. Наиме, како би се уопште прочитала кодна песма неопходно је одређено кретање кроз текст који није линеарно, а разлог томе лежи у природи самих програмских језика који се често пишу као нека врста петље или низа, па са тако не пример нешто у четвртном или петом реду позива на нешто у првом. Постоје овакви случајеви и у језику где се песник позива на нешто што је претходно изречено, али то углавном не спутава нешто линеарније читање песме. Код програмских језика такво враћање у назад готово нужно изискује да се пажња читалаца врати на почетак. Стреховец (2003: 3) бтакође наводи и друге елементе који крше ову линеарност као што су:

читања мишем у смислу да клик на неку реч може отворити или активирати компјутерски програм [...] читањем-посматрање као декодирање због програмабилне природе дигиталног поетског објекта (читалац мора имати на уму који се софтвер користи), перципирање читаве слике налик мозаику на екрану у једном тренутку...

Како би онда заправо изгледала анализа једног дела кодне поезије ако при анализи истог применимо блиско читање. Ако оставимо по страни чињеницу да се блиским читањем може анализирати сваки текст, чак и онај који није нужно уметнички, о чему је Стенли Фиш имао читаве анегдоте односно експерименте, можемо заправо детаљно анализирати сваки елемент кода као и језика и у томе тражити значење. Сходно томе, овај приступ чини се најпогоднијом за анализу кодне поезије управо због инсистирања искључиво на текст, и код у овом случају, као и количини пажње коју изискује за сваки детаљ и траг значења.

Узмимо за пример још једном песму „Црни Бисер” односно “Black Perl”. Сама песма је постављена на јузнет-у (usenet) првог априла 1990-те године (в. Вал, Кристијансен, Орвант 2000: 649) од стране анонимног аутора. Постоје бројне сумње и тврдње да је ауторка заправо Шерон Хопкинс која је у то време имала неколико објављених песама у перл програмском језику, као и један научни рад на ту тему, али је сама негирала да је она аутор тог дела.

На почетку песме у првом стиху имамо команду излаз (exit) због које заправо цео код престаје са извршењем у том тренутку. Аупут, односно крајњи производ овог кода је самим тим ништа, цела се секвенца овде гаси, међутим остатак текст јесте написан смислено и правилно и већина верзија перл програма га разуме⁵, односно, речено језиком информационих технологија, овај код „парсује”, разумљив је самом програму. Треба истаћи да неке новије верзије перл програмског језика не разумеју овај код због одређених промена у самом језику, као и да постоје савременији преводи ове песме која јесте разумљива новим верзијама перл-а, али ту нема великих разлика у књижевном смислу. У новијим верзијама односно, условно речено, преводима где се на неким местима налазило „;“ налази се само зарез, неки зарез је замењен тачком и томе слично.

5 Перл верзија 5 и новије верзије не разумеју ову песму.

Иако на први поглед ово гашење програма на почетку може деловати као грешка, када се анализира остатак текста заправо се оно може и протумачити као врло симболично. Ако погледамо сам крај песме, имао команду чекај, односно „wait”, због начина на који је написана сам програм у том тренутку почиње да чека следеће инструкције које не долазе, сав текст након те команде, која се заправо понављањем додатно наглашава, као и да нису ту што се логике програмског језика тиче. Самим тим се заправо ово чекање у бесконачност и до смрти заправо рефлектује и на почетку само кода, где такође, парадоксално, почиње крај и бесконачно чекање. Овим се разбија линеарност читања и ствара петља о којој професор Страховец (2003: 1–3) пише и о чему је претходно већ било речи.

Остатак текста наставља овом тмурном тематиком коју најављују први и последњи стихови односно редови. Песма говори о пакувима које треба исецкати, убити, поделити и томе слично, оно што је нарочито занимљиво јесте да су све ове обојене речи заправо команде, односно кључне речи које се користе приликом програмирања у перл језику, док сав текст који није обојен више припада људском језику. Дакле у овом случају се емфаза не врши пуким шареним бојама на екрану већ и контрастом у значењу односно програмском разумевању речи. Уједно је на истим тим местима и нагласак приликом читања.

Већ у наредном редовима набрајају се нове жртве, овог пута под знацима навода, што нас наводи на тумачење да се ипак можда не ради о животињама, такође се и наставља игра речи између програмског и људског језика, стихови „помери морал” и „вредност, по страну” (shift moralities, values aside) пружају веома сажет поглед у психолошку свест лирског субјекта, али управо у таквој сажетости и ригидности која је додатно појачана језиком машина огледа се и немилосрдност према споменутиим „жртвама”, како се у самој песми наводи. Још један детаљ нас такође наводи на то да ове жртве нису животиње већ да их поменути лирски субјект само тако доживљава, а то је реч тело односно body. Тело са једне стране има буквално значење тела у материјалном смислу и може указивати на људско тело, али је овде једина реч дата плавом бојом, што још једном није из чисто естетских разлога већ у програмском смислу „body” овде, као и у html-у значи комплетан текст који је ту присутан. Самим тим то нас може довести до тумачења да се заправо кроз читав текст ради о људским телима, односно да се ради о убиству.

Ако се таква интерпретација узме као валидна, онда се заправо добија још један слој саме песме а то је њена намена, тј. цела песма се може схватити као нека врста признања једног убице о својим делима, које своје признање крије не само метафорама и стилским фигурама у песми већ и програмским језиком преко свега тога. Наравно, ово је само једна интерпретација и у духу блиског читања могуће је наставити из потпуно другачијег угла или додатна значења тражити у заградама, резезима и слично. Међутим, и вез даљег анализирања постаје евидентан одговор на питање која је сврха кодне поезија, односно због чега се пишу песме у програмским језицима. Да ли се ради о демонстрацији вештине програмирања или у томе може лежати нешто више књижевно, поетски?

Функција кодној поезији није само украс и естетски ефекта, већ она може садржати у себи значења која додатно обогаћују интерпретацију једне песме. Где је људски језик ограничен симболима и стилским фигурама, мешањем програмских језика долази се до новог начина преношења идеја и порука. Могу се користити команде, боје, секвенце, функције и друге ствари које људски језици немају. Формалисти би ово можда назвали само новим начином онеобичавања текста,

али оно што је суштина овакве анализе јесте проналажење богатства скривених, слојевитих значења у песмама које су по својој природи тешко разумљиве због тога што захтевају познавање и књижевности и програмирања. Иако је, као што смо показали, тешко прецизно дефинисати и одредити шта чини кодну поезију, не треба никако одбацивати овакав вид поезије који је већ неко време на маргини јер у себи крије неистражене или барем слабо истражене механизме песништва.

Литература

Вал и др 2000: L. Wall et al, *Programming Perl*, 3rd Edition, O'Rilley Media, Incorporated.
Rajan 2009: M. Ryan, *Avatars of Story*, Volume 17, Minnesota: University of Minnesota press.

Онлајн литература

Стреховец 2003: J. Strahovec, Text as a Loop/ On the Digital Poetry, <https://www.poetikhars.com/files/deneme/text_as_a_loop.pdf>, 4. 10. 2023.
Tennyson, Alfred, Crossing the Bar, *Poetry Foundation*, <<https://www.poetryfoundation.org/poems/45321/crossing-the-bar>>, 6. 10. 2023.
Water, *Code poetry*, <<https://code-poetry.com/water>>, 8. 10. 2023.
What is black-perl, *Codedocs*, <https://codedocs.org/what-is/black-perl>, 8. 10. 2023.
Source code poetry, <<https://www.sourcecodepoetry.com/>>, 6. 10. 2023.

THE PEARL AND THE AVATAR – CLOSE READING OF CODEWORK POETRY

Summary

The goal of this article is to analyze what precisely constitutes codework poetry, display the problems and complications that arise when attempting to define code poetry and create a categorization of the songs written in programming languages based primarily on the theoretical work of Marie-Laure Ryan titled "Avatars of Story". The article will also strive to modernize and complete her theoretical theories about code poetry, keeping in mind the newly created programing languages and forms of codework poetry. Finally, it will also deal with the possibility of applying the close reading method of literary criticism to the poem "Black Perl" and in doing so explore what the primary function of intermingling human and computer languages within a song is, show how that combination of languages deepens the emotional effect and meaning of poems.

Keywords: code poetry, code, close reading, Marie-Laure Ryan, programming language, metalepsis

Stefan S. Pavić