

# Softwarové umenie optikou teórie znaku a kódu. Počítačový kód ako nositeľ kultúrneho významu.

**Monika Szűcsová**

## **Anotácia**

*Softwarové umenie ako umelecká disciplína má mnoho komunikačných stratégií, ktorými sa snaží o poukázanie na estetiku vlastného umeleckého materiálu, ktorým je počítačový kód. Predkladaný text uvažuje o softwarovom umení, nahliadanom ako literárny útvar, ktorý je možné čítať a interpretovať ako poetický konštrukt. Povrchovou analýzou niektorých z kánonických diel softwarového umenia patriacich do kategórií „codeworks“ alebo kódová poézia, sa zameria na spôsoby, akými sprostredkováva komunikáciu medzi užívateľmi a činiteľmi, ktoré vstupujú do procesu tvorby a prezentácie umeleckého diela. Na dané diela nahliada optikou sémiotiky ako vedy o znakoch, skúmajúcej ako sa objekty či obrazy stávajú nositeľmi významu.*

## **Summary**

*Software art as an artistic discipline uses a wide range of communicational strategies through which it attempts to point out to aesthetics of own artistic material, which is code. Presented text thinks of software art as a literary work that can be read and interpreted as a poetic construct. Surface analysis, some of the canonical software artworks falling into the category of codeworks will focus on ways of how software artwork mediate communication between users and agents who participate in its creation process and presentation. These artworks are viewed from the perspective of semiotics as the science about signs investigating how objects or images take on meaning.*

**Kľúčové slová:** znak, interpretácia, (počítačový) kód, softwarové umenie, jazyk softwaru, sémiotika

**Keywords:** sign, interpretation, (computer) code, software art, language of software, semiotics

## Úvod

Súčasná etapa vnímania a myslenia o digitálnej kultúre a umení je etapou myslenia v jazyku počítačových algoritmov a kódov, ich procesualnosti. Algoritmy môžeme z pohľadu fungovania systému chápať ako postupy určené k riadeniu počítačových procesov a dát. Jazykom digitálnych médií je kód, majúci v určitej etape kultúrneho vývoja v rámci určitej komunity nejaký význam.<sup>1</sup> Tento dáva danej komunite alebo aj mimo komunitu zmysel v momente, keď jej je známy jazyk, akým bolo dielo „napísané“, dochádza k odkrytiu jeho významu. Na základe tohto určenia je daná skupina schopná dielo pochopiť, interpretovať ho spôsobom ako to autor zamýšľal, pomôcť k dekodovaniu umeleckého diela alebo jeho procesu.

Predkladaný text sa pokúsi uviesť do kontextu významové štruktúry jazyka a „pravopisu“ vybraných kánonických diel umeleckej kategórie označovanej ako softwarové umenie<sup>2</sup>, jeho estetické možnosti a pohnútky v rámci kreatívnej praxe tvorby, ktorej materiálom je počítačový kód. Zdrojový kód počítača je textovou formou programovacieho kódu, preto o ňom môžeme uvažovať ako o literárnom, textovom útvaru, ktorý je možné čítať a interpretovať ako poetický konštrukt. Zároveň sa optikou sémiotiky, ako vedy o znakov skúmajúcej ako sa objekty či obrazy stávajú nositeľmi významu, zameria na povrchovú analýzu znakov kultúry, ktorá je determinovaná digitálnou technológiou. V medziach tohto vzťahu bude skúmať umeleckú tvorbu, nahliadanú cez počítačový kód ako kultúrny artefakt a spôsoby, ktorými je táto tvorba vnímaná a interpretovaná.

Softwarové umenie je v pojatí súčasných teoretikov a umelcov digitálnych médií chápané ako činnosť, ktorá sa snaží o artikuláciu širokej škály vlastností softwaru (kód ako poézia, konceptuálny software, kód ako plagiátorstvo alebo koláž a podobne) v kontexte kreatívnej tvorby, ako aj o jeho reflexiu v rámci kultúry.<sup>3</sup> Proces tvorby zahŕňa projekty, používajúce software ako hlavný umelecký materiál, pričom môžeme generatívne kódy považovať za inovatívnu zložku umeleckého procesu. Počítačový programovací kód ale nie je hodnotovo neutrálny a nemožno na neho poeticky nahliadať bez kritickej analýzy a posudzovania, uváženia jeho možností a vplyvu – či už na kultúru, umenie, spoločnosť,

---

<sup>1</sup> Viz Kittler, Friedrich. Code (or, How You Can Write Something Differently). In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008. Manovich, Lev. *Jazyk nových médií*. Praha: Karolinum, 2018.

<sup>2</sup> Softwarové umenie ako kreatívny proces v prostredí programovateľných médií chápem v zmysle ako ho definovala porota festivalu *transmediale. 01* v Berlíne (2001) – v tom čase ako definíciu súťažnej kategórie „umelecký software“: „Toto ocenenie je o algoritme; jedná sa o kód, ktorý generuje, spracováva a spája to, čo môžete vidieť, počuť a cítiť. [...] Softwarové umenie by mohli byť algoritmy ako koniec samých seba, mohlo by rozvrátiť pochopené paradigmy počítačového softwaru alebo vytvoriť nové, mohlo by urobiť niečo zaujímavé alebo rušivé s vaším počítačom, mohlo by byť tvorivým písaním, mohlo by byť vedou. [...] Vzhľadom k tomu, že hovoríme o algoritmickom kóde, keď hovoríme o softwarovom umení, hovoríme zároveň o programovacích jazykoch - možno dokonca o poézii v programovacích jazykoch - a hovoríme o rozdiel medzi zdrojovým kódom v programovacích jazykoch a realizovateľným zostaveným kódom.“

Zdroj: Artistic software jury. Award - artistic software. *Transmediale.01*. [online] 2001 [cit. 10. prosince 2019]. Dostupné z <[http://pastwebsites.transmediale.de/01/en/s\\_juryStatement.htm](http://pastwebsites.transmediale.de/01/en/s_juryStatement.htm)>.

<sup>3</sup> Fuller, Matthew (ed.). *Software Studies: A Lexicon*. Cambridge, Massachusetts: MIT Press, 2008.

prípadne výchovu. Podobne ako iné formy umeleckej tvorby, aj softwarové umenie nevyhnutne potrebuje súbor lingvistických pravidiel, znakov a kľúča k ich kombinovaniu, aby sme pochopili prostredníctvom čoho komunikuje s užívateľmi a činiteľmi, ktoré vstupujú do procesu tvorby a prezentácie. Nástrojmi k pochopeniu a interpretácii daných kultúrnych a jazykových znakov v rámci umeleckej praxe sa môže stať teória sémiotiky<sup>4</sup>, reflexia umeleckej tvorby v dobe determinovanej digitálnymi technológiami<sup>5</sup>, analýza významu programového kódu v umení, ktorého materiálom a jazykom je software.<sup>6</sup>

### **Jazyk softwaru: kód, znak, interpretácia**

Digitálne médiá sú v súčasnej kultúre nositeľmi určitej autonómie, na ktorú upozorňuje teoretik médií Friedrich Kittler.<sup>7</sup> Tento teoretik zároveň považuje software za kultúrnu prax, ktorej jednotlivé vrstvy musíme študovať. Poukazuje na nevyhnutnosť porozumieť kódom ako jazyku našej doby, v opačnom prípade nám hrozí, že sa v tejto kultúrnej vrstve stratíme. Naša kultúra a existencia sú dnes plne determinované kódmi, ktoré nás obklopujú. Možnosti kódovania (a dekódovania) ovplyvnil podľa neho už vznik gréckej abecedy, ktorá predstavuje síce nedostatočný (z dôvodu nedostatočne vyvinutých komunikačných technológií), napriek tomu však zásadný predpoklad ku kódovaniu.

Kittler ďalej zdôrazňuje myšlienku vplyvného matematika Alana Turinga, že z povahy samotných počítačov vyplýva, že pracujú na báze kódov, sú teda schopné dekódovať a preložiť neznáme kódy a ich významy.<sup>8</sup> Zároveň Kittlera trápi dilema medzi kódom a prirodzeným ľudským jazykom, ktorú, obáva sa, nemožno vyriešiť. Tvrdí, že buď môžeme skúsiť sami (programátori) napísať kód, ktorý bude schopný odhaľovať svoju podstatu a stane sa pre nás čitateľným (čoho dôsledkom by zrejme bolo nespočetné množstvo kódov generovaných informácií a finančná náročnosť takejto operácie), na druhej strane môžeme nechať počítače, aby si odvodili (napísali) kódy zo svojho vlastného prostredia,

---

<sup>4</sup> Eco, Umberto. *O zrcadlech a jiné eseje: znak, reprezentace, iluze, obraz*. Praha: Mladá fronta, 2002. Eco, Umberto. *Teorie sémiotiky*. Brno: Janáčkova akademie múzických umění v Brně, 2004. Mukařovský, Jan. Umění jako sémiologický fakt (1934). In *Studie I*. Mukařovský, Jan. Brno: Host, 2000.

<sup>5</sup> Kesner, Ladislav. *Muzeum umění v digitální době. Vnímání obrazů a prožitek umění v soudobé společnosti*. Praha: ARGO a NG, 2000.

<sup>6</sup> Cramer, Florian - Gabriel, Ulrike. *Software Art*. [online] 2001 [cit. 10.prosinec 2019]. Dostupné z <[http://www.netzliteratur.net/cramer/software\\_art\\_-\\_transmediale.html](http://www.netzliteratur.net/cramer/software_art_-_transmediale.html)>. Cramer, Florian. *Digital Code and Literary Text*. [online] 2001 [cit. 10.prosinec 2019]. Dostupné z <[http://www.netzliteratur.net/cramer/digital\\_code\\_and\\_literary\\_text.html](http://www.netzliteratur.net/cramer/digital_code_and_literary_text.html)>. Cramer, Florian. *When writing executes itself. Poetry and Program Code*. [online] 2003 [cit. 10.prosinec 2019]. Dostupné z <[http://cramer.pleintekst.nl/all/wenn\\_schrift\\_sich\\_selbst\\_ausf%FCht/writing\\_executing\\_itself.rtf](http://cramer.pleintekst.nl/all/wenn_schrift_sich_selbst_ausf%FCht/writing_executing_itself.rtf)>. Cramer, Florian. *Words Made Flesh: Code, Culture, Imagination*. Rotterdam: Willem de Kooning Academy Hogeschool, 2005. Cramer, Florian. Language. In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008.

<sup>7</sup> Kittler, Friedrich. Code (or, How You Can Write Something Differently). In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008. Kittler, Friedrich. There is no software. *CTheory.net*. [online] 18. října 1995. Dostupné z <<http://www.ctheory.net/articles.aspx?id=74>>.

<sup>8</sup> Kittler, Friedrich. Code (or, How You Can Write Something Differently). In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008.

tieto však zostanú pre nás nečitateľné, to znamená, že ich nebudeme schopní pomocou prirodzeného jazyka artikulovať.

Prirodzený, ľudský jazyk predstavuje v prostredí počítačov dáta, ktoré sú spracovávané softwarom. V tomto kontexte uvádza Florian Cramer delenie formálneho jazyka v softwari na dve úrovne: jazyk, ktorý sa realizuje vo vnútri softwaru a jazyk, v ktorom je software napísaný (programovací jazyk).<sup>9</sup> Programovací jazyk predstavuje syntaktickú manipuláciu symbolov. Jeho operačná sémantika sa rozchádza s tradičným lingvistickým chápaním jazyka a z tohto pohľadu je aj často nezmyselná. Na druhej strane, zdrojové kódy počítačového programu ako sady inštrukcií napísané programovacím jazykom vyššej úrovne, sú prostredníctvom prirodzeného jazyka čitateľné, pretože používajú jeho určité charakteristiky (pravidlá syntaxe). Počítačová kultúra je súborom čísiel a znakov a samotný počítačový kód pôsobí v rámci určitej komunity ako nositeľ významu, jazyka a ideológie daného kolektívu alebo kultúry. Lev Manovich používa pojem „kultúrne rozhranie“ k popisu rozhrania „človek – počítač – kultúra“ a toto rozhranie chápe ako sémiotický kód informačnej spoločnosti.<sup>10</sup>

Kódy sprostredkovávajú pravidlá, generujúce znaky, ktoré predstavujú konkrétne udalosti v rámci komunikačného styku.<sup>11</sup> Znaky, ako výsledky kódových pravidiel sa rozkladajú v komplikovanej sieti premenlivých vzťahov. Na stránkach svojej eseje *Znaky, ryby a knoflíky. Poznámky k sémiotice, filozofii a humanitným vedomiam* Eco predostiera „dialóg“ medzi Barthesovou analýzou jazyka a ním aplikovaný lingvistický model na kuchársku terminológiu ako znakový a významový systém, a kritiku tejto Barthesovej analógie filozofom Rogerom Scrutonom.<sup>12</sup> Podobne ako je kuchársky recept nositeľom významov a sadou inštrukcií, ktorými sa pri varení musíme riadiť, chápeme počítačové algoritmy ako nositeľov metódy a formálnych príkazov, ktoré musí programátor nasledovať.<sup>13</sup> Softwarový zdrojový kód a kuchársky recept zdieľajú jednu kľúčovú vlastnosť a tou je, že sú nositeľmi informácií. Zdrojový kód počítača je navyše napísaný na úrovni programovacieho jazyka, ktorá je človekom čitateľná (či už sa jedná o jazyk Perl, Python, BASIC a podobne). Krysa a Sedek vo svojom texte *Source Code* uvádzajú príklad projektu *Barszcz.net*, ktorý použitím rôznych programovacích jazykov prepisuje recept na polievku boršč do zdrojového kódu.<sup>14</sup> Sledujú tým myšlienku slobodného

---

<sup>9</sup> Cramer, Florian. Language. In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008.

<sup>10</sup> Manovich, Lev. *Jazyk nových médií*. Praha: Karolinum, 2018.

<sup>11</sup> Eco, Umberto. *Teorie sémiotiky*. Brno: Janáčkova akademie múzických umění v Brně, 2004.

<sup>12</sup> Eco, Umberto. *O zrcadlech a jiné eseje: znak, reprezentace, iluze, obraz*. Praha: Mladá fronta, 2002.

<sup>13</sup> Krysa, Joasia - Sedek, Grzesiek. Source code. In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008.

<sup>14</sup> Ibid.

prístupu ku kódu v rámci kreatívnej kolaborácie užívateľov verejných domén a v neposlednom rade poukazujú na blízkosť zdrojového kódu a kuchárskeho receptu v doslovnom zmysle:<sup>15</sup>

```
/ Barszcz C recipe
*
* string based cooking
*
* Copyleft (C) 2006 Denis "Jaromil" Rojo
* for the barszcz project (currently unfinished)

#include <stdio.h>
#define ingredient char
#define tool char

#define few 3
#define some 5
#define pinch 1
#define plenty 8
#define one 1

#define soft_cooked 5

ingredient **take(int quantity, ingredient **ingr) {
    int c;
    int len = strlen(ingr) +10;
    ingredient = malloc( (quantity+1) * sizeof(*ingredient));
    for(c = 0; c < quantity; c++)
        ingredient[c] = malloc(len * sizeof(ingredient));
    ingredient[c+1] = NULL;
    return ingredient;
}
```

Všeobecná sémiotika je filozofickou činnosťou, znakovým systémom, ktorý pôsobí v oblasti prirodzených jazykov a nezaobera sa výlučne len teóriou znakov: „*Specifické sémiotiky [...] popisujú interpretační systémy, které jsou různými systémy znaků. [...] popisují, organizují mimický jazyk, verbální jazyk, vizuální znakový systém. [...] Obecná sémiotika naopak stanovuje prostřednictvím filozofického gesta sám obecný pojem znaku, právě abychom mohli mluvit jednoduchým způsobem o jevech, jež si na povrchu nejsou nijak podobné. [...] je uvažováním o znaku, o znakovosti nebo o hlubokých mechanismech každého významového systému.*“<sup>16</sup> Rozhranie novomediálneho (programovateľného) umeleckého diela ako systém, sprostredkávajúci význam a prostredie vzniku novej vizuálnej skúsenosti, je agentom nových spôsobov vnímania materiality diela a estetickej

---

<sup>15</sup> Ibid., s. 236.

<sup>16</sup> Eco, Umberto. *O zrcadlech a jiné eseje: znak, reprezentace, iluze, obraz*. Praha: Mladá fronta, 2002, s. 430.

skúsenosti diváka s dielom. Obsah a rozhranie nemôžu byť chápané samostatne, pretože obsah diela determinuje výber rozhrania.<sup>17</sup>

Dekódovanie obsahov umeleckého diela pomocou znakov analyzuje v prvej polovici minulého storočia Jan Mukařovský.<sup>18</sup> Existenciu znakov chápe vo dvoch rovinách: ako kolektívne vedomie a zmyslový svet. Jedným z dôležitých aspektov k interpretácii významu diela je u neho kolektívne vedomie, ktoré obsahuje sumu subjektívnych stavov vedomia členov určitého kolektívu. Dielo do tejto interpretácie vstupuje ako externý činiteľ, nazýva ho symbolom respektíve podľa Saussurovej<sup>19</sup> terminológie pojmom „signifiant“.<sup>20</sup> Obsahy subjektívnych asociácií spojených s estetickým vnímaním, nachádzame v kolektívnom vedomí, do aktu interpretácie diela ale vstupuje aj už zmienená rovina druhá, subjektívne zmyslové vnímanie. Umelecké dielo má v jeho výklade dve funkcie: funkciu autonómneho znaku a znak ako komunikačný prostriedok. Jeho autonómny znak funguje v roli prostredníka v rámci určitej komunity alebo kolektívu, i keď musí pôsobiť a byť pochopený ako vysielateľom, tak prijímateľom obsahu.

Eco predostiera termín interpretácie ako snahu o pochopenie niečoho, čo nie je zrejmé a spája ho napríklad s tradíciou dešifrovania textov.<sup>21</sup> Označuje ju za sémiotický fenomén: „každý znak, každý symbol v širokém smyslu slova je interpretáci. Pojem interpretácie se nemusí nutně ztotožňovat s hledáním temného, skrytého smyslu: „*je to prostředek, jímž nějak objasňujeme nějaký smysl (každý výraz, jehož význam neznáme, je temný)*“.<sup>22</sup> Podstatným atribútom znaku je to, že v procese interpretácie zastupuje sám seba a svoje možné interpretácie; je teda tým, čo môže byť interpretované. Na otázku interpretácie nahliada optikou hermeneutickej tradície. Akt pochopenia určitého stavu uvádza na príklade analýzy literárneho diela, jeho estetických a umeleckých kvalít. Upozorňuje, že k uspokojivému pochopeniu jednotlivých častí textovej informácie, si interpret nevyhnutne musí najprv stanoviť hypotézu, s ktorou bude k dielu pristupovať. Tento stav interpretácie nazýva „pred-pochopením“, vychádzajúcim z predchádzajúcich interpretácií. Do takéhoto „pred-pochopenia“ interpretuje objekt aj spôsob jeho používania. Daný proces sprostredkuje interpretovi pochopenie následnosti jednotlivých faktov, tvoriacich celok.

Majúc na pamäti vyššie zmienenú definíciu hermeneutického kruhu, môžeme tvrdiť, že poučený (znalý) interpret softwarového umeleckého diela pristupuje k takémuto dielu so sumou

---

<sup>17</sup> Manovich, Lev. *Jazyk nových médií*. Praha: Karolinum, 2018.

<sup>18</sup> Mukařovský, Jan. *Umění jako sémiologický fakt (1934)*. In *Studie I*. Mukařovský, Jan. Brno: Host, 2000.

<sup>19</sup> Viac o Saussurovej teórii znakov napríklad v texte: Holdcroft, David. *Saussure: signs, system, and arbitrariness*. New York: Cambridge University Press, 1991.

<sup>20</sup> Ibid.

<sup>21</sup> Eco, Umberto. *O zrcadlech a jiné eseje: znak, reprezentace, iluze, obraz*. Praha: Mladá fronta, 2002.

<sup>22</sup> Ibid., s. 423.

predpokladov a znalostí a v konfrontácii s dielom samotným očakáva naplnenie svojich domnienok, teda že sa dielo nejakým spôsobom „zachová“, niečo „urobí“, niečo „vyvolá“; a to tak v interpretácii ako v publiku. Takúto prípravu na stretnutie s umením neočakávame samozrejme len v prípade diel softwarového umenia. Predovšetkým za posledných sto rokov sme boli svedkami mnohých konceptuálnych umeleckých hnutí (dadaizmus, Fluxus, surrealizmus a podobne), ktoré predpokladali určitý „preklad“ jazyka svojej tvorby, nutný k pochopeniu jeho významov.

### **Niektoré zo spôsobov využitia softwaru ako materiálu kreatívnej praxe**

Počítačový kód pozostáva podobne ako iné jazyky z jemu vlastných pravidiel respektíve syntaxe a významovej zložky, teda sémantiky. Môže sa prepisovať do rôznych významov, či už sa jedná o vyjadrenie matematických operácií alebo poézie. Programovateľné dielo môžeme čítať ako literatúru, ktorej abecedou je sústava jednotiek a núl. Cramer a Gabriel vytvárajú paralelu medzi definíciou literatúry, ktorá pozostáva z písmen alfanumerickej abecedy, čitateľnej človekom a literatúry, ktorú tvoria softwarové protokoly a programové kódy.<sup>23</sup> Syntax, ktorou je počítačový kód napísaný, pozostáva z mnohých úrovní kódu, ktorý sa v momente spustenia neustále prepisuje do znakov vyššej úrovne, ktoré sú čitateľné človekom.

Používanie softwaru ako materiálu umeleckej tvorby v prostredí programovateľných médií má mnoho úrovní. Či už sa jedná o kódové diela typu „codeworks“ Alaina Sondheima, Grahama Harwooda, anonymnej identity na internete nazvanej Netochka Nezvanova alebo ASCII kódov, známych predovšetkým z diel Vuka Ćosića, ďalej diela poukazujúce na internetové rozhranie počítača (*Webstalker*, 1997) a odkazujúce ku kódu na jeho pozadí (tvorba JODI), tak aj projekty pracujúce s kódom ako výkonným činiteľom. Kurátorka a teoretička nových médií Inke Arns je fascinovaná generatívnym potenciálom softwarového kódu a vníma ho ako nástroj, ktorého záujmom je uskutočňovanie alebo realizácia nejakého úkonu v rámci techno – sociálneho systému.<sup>24</sup> Za najradikálnejší príklad užívania počítačového kódu ako umeleckého materiálu považuje tvorbu pod názvom „codeworks“. Tento typ tvorby si kreatívnym spôsobom privlastňuje programovací jazyk a počítačový kód. Pripomína myšlienku Floriana Cramera (2001), že počítač nie je obrazové, ale textové médium (skladajúce sa z dvoch úrovní: fenotextu, ktoré sú na povrchu a komunikujú s užívateľom a genotextu, ktorými sú zdrojové texty a programové kódy), ku ktorému je možné pripojiť rôzne audio-vizuálne výstupné zariadenia.

---

<sup>23</sup> Cramer, Florian - Gabriel, Ulrike. *Software Art*. [online] 2001 [cit. 10. prosince 2019]. Dostupné z <[http://www.netzliteratur.net/cramer/software\\_art\\_-\\_transmediale.html](http://www.netzliteratur.net/cramer/software_art_-_transmediale.html)>.

<sup>24</sup> Arns, Inke. *Read\_me, run\_me, execute\_me*. Code as Executable Text: Software Art and its Focus on Program Code as Performative Text. In *Medien Kunst Netz 2: Thematische Schwerpunkte*. Frieling, Rudolf – Daniels, Dieter (eds.). Wien - New York: Springer, 2005.

Pre ilustráciu vyššie zmieneného uveďme príklad net.artového umelca Vuka Ćosića, ktorý v roku 1995 zaviedol pojem net.art.<sup>25</sup> Ćosić sa v umeleckej praxi venuje tvorbe ASCII (American Standard Code for Information Interchange) filmov, odhaľujúcich počítačový kód a tým odkazujúcich k povahe digitálnych médií ako počítačových dát (Manovich, 2018).<sup>26</sup> ASCII abeceda predstavuje súbor znakov, definovaných štandardom ASCII z roku 1966 a predchodcov je možné dohľadať už v starovekom Egypte, pri prepisovaní symbolov do obrázkov. Novšia história používania ASCII kódov siaha do 60. – 70. rokov 20. storočia, keď sa začal používať v počítačovom prostredí z dôvodu limitovaných možností počítačov pre znázornenie určitých symbolov. Podľa Manovicha odkazuje používanie ASCII kódu ku kategóriám kryptografie alebo systémom kódovania a Ćosićovi sa podarilo do svojich filmov „zakódovať“ mnoho zásadných prvkov novomediálneho umenia a počítačovej kultúry.<sup>27</sup> Arns nazýva akt uprednostňovania programového kódu pred povrchom obrazovky a genotextu pred fenotextom, oslobodzujúcim efektom, ktorý prináša tvorba v prostredí ASCII kódov.<sup>28</sup> Zároveň poukazuje na prínos, aký majú kódové diela na uvedomovanie si narastajúcej programovateľnosti nášho mediálneho prostredia.

Artikuláciu možností kódovej estetiky v oblasti literatúry môžeme nájsť už v 70. rokoch 20. storočia v diele Alana Sondheima. Okrem toho, že zaviedol pojem „codeworks“ (kódové diela), tak experimentoval s programovacím kódom príkazového riadku, ktorý chápal v širšom kontexte konceptuálneho umenia. V Sondheimovom diele sme svedkami prepájania a kombinovania syntaxu a sémantiky, subjektu a objektu, formalizmu a kultúry.<sup>29</sup> Odhaľuje počítačový kód, ktorý sa v jeho diele mieša a zlučuje s prirodzeným jazykom.

Poéziou tvorenou v programovacom jazyku Turbo Pascal sa už počas svojich štúdií zaoberal teoretik nových médií a umelec Noah Wardrip-Fruin.<sup>30</sup> Vytvoril generátor poézie, ktorý bol zameraný na proces tvorby viet. V tomto prístupe autor zámerne zviditeľňuje proces písania a proces tvorby (teda prácu softwaru). Neskôr sa Wardrip-Fruin zaoberá kreatívnymi možnosťami hry s jazykom a používaním

---

<sup>25</sup> Stalo sa tak celkom náhodne, v momente keď obdržal email od anonymného odosielateľa a z dôvodu nekompatibilitnosti softwaru, dokázal rozlúštiť len fragment emailu, ktorý odkazoval k prirodzenému jazyku a dával zmysel: [...] J8~g#|;Net. Art{-^s1 [...]. Ćosić sa ku správe postavil vynaliezavo a jediné čitateľné slovo textu: *Net. Art* začal používať ako názov umeleckej aktivity, ktorej sa v prostredí internetu venoval. Obdržaný email však Ćosić neskôr preposlal k dekodovaniu kolegovi Igorovi Markovicovi, aby sa dozvedel, že email bol vágnym a kontroverzným manifestom tradičných umeleckých inštitúcií a volaním po slobode vyjadrovania v prostredí internetu.

<sup>26</sup> Manovich, Lev. *Jazyk nových médií*. Praha: Karolinum, 2018.

<sup>27</sup> Ibid.

<sup>28</sup> Arns, Inke. Read\_me, run\_me, execute\_me. Code as Executable Text: Software Art and its Focus on Program Code as Performative Text. In *Medien Kunst Netz 2: Thematische Schwerpunkte*. Frieling, Rudolf – Daniels, Dieter (eds.). Wien - New York: Springer, 2005.

<sup>29</sup> Cramer, Florian. *Words Made Flesh: Code, Culture, Imagination*. Rotterdam: Willem de Kooning Academy Hogeschool, 2005.

<sup>30</sup> Wardrip-Fruin, Noah. *Expressive Processing. Digital Fiction, Computer Games, and Software Studies*. Cambridge – London: MIT Press, 2009.



počítačových procesov, ktoré sú v dnešnej kultúre a umení umožnené digitálnymi technológiami.<sup>31</sup> Počítačové procesy sú nástrojom k vytváraniu nových modelov jazyka digitálnych technológií, ktorý ale poskytuje porozumenie len „čitateľom“, ktorí rozumejú počítačovým procesom. Do svojej tvorby zavádza nové formy textovej logiky a demonštruje to na príklade projektu *Two n-gram Instruments* (2004), ktorý pracuje s lingvistickým modelom práce s ľudským jazykom, nazývaným Markovova reťaz. Príklady Markovovej reťaze alebo Backus–Naurovej formy<sup>32</sup> označuje Kittler za vhodné spôsoby, akými by sa mali kódy, ako konečné sekvencie symbolov, byť schopné nekonečne reprodukovat'.<sup>33</sup> Wardrip-Fruinova jazyková hra v textovom kontexte *Two n-gram Instruments* je príkladom práve takéhoto množenia textu. Textové inštrumenty, ktoré v rámci tohto projektu vytvoril, pracujú na princípe reťazenia slov pôvodného textu, pochádzajúceho z rôznych častí zdrojového dokumentu a interakcie okien prehliadača, čoho dôsledkom je vznik nových textových obsahov. Takto vygenerované textové štruktúry umožňujú hravé interakcie s textom (vyžadujú interakciu a majú performatívny charakter) a nové spôsoby objavovania jazyka.

Predpoklad performancie zdieľa s vyššie uvedeným aj softwarové dielo, prepisujúce sa priamo do fyzického priestoru kde vyžaduje byť realizované. Tvorcom *dot.walk* (2004), open source generatívnej psychogeografie je Wilfried Hou Je Bek (projekt Socialfiction.org). Napísané algoritmy počítača tu slúžia ako inštrumenty k navigácii po meste a vyzývajú recipientov diela, aby sa zúčastnili tohto procesu:<sup>34</sup>

```
// Classic.walk

Repeat

{

1 st street left
2 nd street right
2 nd street left

}
```

Proces realizácie diela spočíva v nasledovaní vygenerovaného algoritmu, teda účastník má ísť v jednej ulici doľava, potom zase zahnúť doprava, celý proces zopakovať a podobne. Algoritmus je

---

<sup>31</sup> Ibid.

<sup>32</sup> Backus–Naurova forma je spôsob zápisu bezkontextových gramatik používaných pre popis formálnych jazykov, napríklad k popisu syntaxe počítačového programovacieho jazyka ALGOL.

<sup>33</sup> Kittler, Friedrich. Code (or, How You Can Write Something Differently). In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008.

<sup>34</sup> Cramer, Florian. *When writing executes itself. Poetry and Program Code*. [online] 2003 [cit. 10.prosinec 2019]. Dostupné z <[http://cramer.pleintekst.nl/all/wenn\\_schrift\\_sich\\_selbst\\_ausf%FCht/writing\\_executing\\_itself.rtf](http://cramer.pleintekst.nl/all/wenn_schrift_sich_selbst_ausf%FCht/writing_executing_itself.rtf)>.

teda čitateľný na úrovni bežného jazyka a čitateľovi odkrýva svoj význam. Obržaná „mapa“ je algoritmom napísaná trasa prechádzky, je v podstate jeho realizáciou vo forme fyzického pohybu po meste. Zápis je samozrejme často nezmyselný v zmysle, že je fyzicky nerealizovateľný, zostáva teda na úrovni konceptuálnej pohnútky.

Cramer spochybňuje tézu o všeobecnej domnienke, že software nie je estetický, ale funkčný nástroj.<sup>35</sup> Je schopný vytvárať esteticky hodnotné umelecké výstupy, môže pôsobiť pochybnosť, vyzýva k účasti na svojom procese, má vlastnú rétoriku a štýl. Je materiálom umeleckej subjektivity, s ktorým umelci vedome pracujú. Ďalej uvádza, že je chybné sa domnievať, že strojový jazyk je čitateľný len pre stroje a teda irelevantný pre ľudskú literatúru a umenie a naopak, že literatúra a umenie by nesúviseli s formálnym jazykom strojov. Literatúra by nás mala naučiť, pochopiť textualitu počítačov a digitálnej poézie a na strane druhej by nás mali počítače a digitálna poézia naučiť ako venovať viac pozornosti kontrolným štruktúram, ktoré sú zakódované do všetkých jazykov<sup>36</sup>.

### Poetická estetika programového kódu

Vzhľadom k tomu, že sú digitálne technológie založené na sústave jednotiek a núl, môžeme ich prekladať do akýchkoľvek jazykov bez straty informácií v procese prekladu. Cramer vyzdvihuje privilegovanosť literatúry oproti napríklad hudbe alebo vizuálnemu umeniu pri preklade tejto symbolickej formy digitálneho informačného systému do digitálneho kódu.<sup>37</sup> Dôvodom je, že literatúra už kódom je, kdežto obrazy a zvuk sa do kódu musia prekladať, to znamená, že dochádza k prekladu nesémantického diela (pixely) do sémantického (slovný opis). Zároveň Cramer definuje počítačovú alebo digitálnu poéziu ako literatúru, pre distribúciu ktorej nie je za potreby konkrétne médium, teda počítač alebo počítačová sieť, pretože môže byť rovnako tak existovať vo forme tlačenej knihy alebo sa generovať v mysli čitateľa.<sup>38</sup>

Prvé poetické texty, ktoré boli napísané v programovacom jazyku počítača sa datujú do začiatku 70. rokov 20. storočia, keď sa objavujú v tvorbe Oulipo (**O**uvroir de **l**ittérature **p**otentielle). Člen tohto združenia potenciálnej literatúry, matematik François le Lionnais, ktorý je autorom prvej takejto básne ju napísal v programovacom jazyku ALGOL.<sup>39</sup> Kým ALGOL bol populárny predovšetkým v 60. a 70.

---

<sup>35</sup> Cramer, Florian. *Digital Code and Literary Text*. [online] 2001 [cit. 10. prosince 2019]. Dostupné z <[http://www.netzliteratur.net/cramer/digital\\_code\\_and\\_literary\\_text.html](http://www.netzliteratur.net/cramer/digital_code_and_literary_text.html)>.

<sup>36</sup> Časť Cramerovej teórie uvedenej v tomto odstavci zaznela v mojej magisterskej diplomovej práci: Szűcsová, Monika. (2016) *Softwarové umění: genealogie, historie, jazyk*. Magisterská diplomová práce obhájená na TIM, FF MU v roce 2016. Vedoucí: doc. Jana Horáková, 2016. Dostupné z <[https://is.muni.cz/th/384256/ff\\_m?info=1;zpet=%2Fvyhledavani%2F%3Fsearch%3DSZ%C5%B0CSOV%C3%81%26start%3D1](https://is.muni.cz/th/384256/ff_m?info=1;zpet=%2Fvyhledavani%2F%3Fsearch%3DSZ%C5%B0CSOV%C3%81%26start%3D1)>.

<sup>37</sup> Ibid.

<sup>38</sup> Ibid.

<sup>39</sup> Cramer, Florian. *When writing executes itself. Poetry and Program Code*. [online] 2003 [cit. 10. prosince 2019]. Dostupné z <[http://cramer.pleintekst.nl/all/wenn\\_schrift\\_sich\\_selbst\\_ausf%FCht/writing\\_executing\\_itself.rtf](http://cramer.pleintekst.nl/all/wenn_schrift_sich_selbst_ausf%FCht/writing_executing_itself.rtf)>.

rokoch minulého storočia, začiatkom 90. rokov sa poéziou napísanou za pomoci programovacích jazykov, začína zaoberať komunita, používajúca programovací jazyk Perl (Practical Extraction and Report Language).

Tvorcom jazyka Perl a zároveň zakladateľom Perl poézie je Larry Wall, ktorý v roku 1999 obhajuje vhodnosť jazyka Perl pre postmodernú dobu. Ďalej dodáva, že je jazykom kultúry, ktorý má konvencie a programovanie v tomto jazyku predstavuje žánrové programovanie; teda zároveň s vývojom programovacieho jazyka bola vyvinutá kultúra, ktorá ho obklopuje a ktorú Perl definuje.<sup>40</sup> Cramer (2003) poukazuje na jav, ktorý je bežný u rôznych umeleckých žánrov v prostredí digitálnych technológií, že najzaujímavejšie príklady digitálnej poézie boli napísané net.artovými umelcami a nie spisovateľmi.<sup>41</sup>

Nakoniec literárne diela spadajúce do smeru „Perl poetry“ nezaznamenali väčší úspech, jednou z výnimiek je snáď len tvorba Grahama Harwooda. Umelec, tvoriaci v prostredí digitálnych médií, prepísal v roku 2001 klasickú anglickú báseň Williama Blakea *London* (1791) do programovacieho jazyka Perl. Námetom Blakeovej básne bolo umieranie chudobných detí v expandujúcom Londýne druhej polovice 18. storočia. Segment Harwoodovho prepisu *London.pl* (2001) vyzerá nasledovne:<sup>42</sup>

```
#      Find and calculate the gross lung-capacity of the
children
#      screaming from 1792 to the present
#      calculate the air displacement needed to represent
the public
#      scream
#      set Public Address System instance and transmit the
output.
#      to do this we approximate that there are 7452520 or
so faces
#      that live in the charter'd streets of London.
#      Found near where the charter'd Thames does flow.
```

Harwood sa v básni realizuje na dvoch úrovniach, na úrovni literatúry a na úrovni fyzického priestoru (počítačový kód ako prostriedok ku komunikácii, realizujúci sa v prostredí digitálnych médií ako proces). Výstup jeho básne je zaujímavý tak z hľadiska básnickej poetiky ako aj z pohľadu zdrojového kódu, teda ako vygenerovaný text (fenotext) a príčina, dôvod vzniku textu (genotext).<sup>43</sup>

---

<sup>40</sup> Wall, Larry. *Perl, the first postmodern computer language*. [online] 1999 [cit. 10.prosinec 2019]. Dostupné z <<https://cseweb.ucsd.edu/~goguen/courses/230/pm-perl.html>>.

<sup>41</sup> Cramer, Florian. *When writing executes itself. Poetry and Program Code*. [online] 2003 [cit. 10.prosinec 2019]. Dostupné z <[http://cramer.pleintekst.nl/all/wenn\\_schrift\\_sich\\_selbst\\_ausf%FCht/writing\\_executing\\_itself.rtf](http://cramer.pleintekst.nl/all/wenn_schrift_sich_selbst_ausf%FCht/writing_executing_itself.rtf)>.

<sup>42</sup> Ibid., nestr.

<sup>43</sup> Ibid.

Festival v Dortmunde *Temporary Software Art Factory Readme 100*, 2005) uviedol softwarové dielo Ilyu Malinovskoho *LYCAY (Let Your Code pLAY, 2005)*, ktorého predmetom záujmu je „preklad“ pravidiel jazykovej gramatiky do hudby. Pri prevode formálneho jazyka do zvuku zostáva zachovaný význam výpovede, mení sa len médium. Autor v tomto diele používa nasledovné matematické formalizmy: „akýkoľvek programovací jazyk je organizovaný svojou formálnou gramatikou. Význam formálnej gramatiky je systém funkcií, závislých na gramatických pravidlách jazyka a jadrových pravidlách (tj. konkrétne výrazy: názvy premenných, premenné hodnoty a tak ďalej). Ak existujú spôsoby ako rozozvučať všetky pravidlá gramatiky, potom zvuk každého pravidla zodpovedá významu tohto pravidla, takže zvuk celého programu tiež zodpovedá významu programu.“<sup>44</sup> Gramatické pravidlá programovacieho jazyka JAVA prekladá do hudobných fráz a tým dochádza k reprezentácii programu v hudbe. Proces programovania doprevádza hudba, korešpondujúca s významom programu a slúžiaca k vyjadreniu estetickéj krásy programovacieho kódu.

Jedným z kánonických diel Perl poézie je minimalistická umelecká partitúra Floriana Cramera *Self.pl* z roku 2001.

```
#!/usr/bin/perl
open (IT, "< self");
while (<IT>) {
push @it, $_}
close (IT);
open (IT, ">> self");
print IT join ("\n ", @it);
close (IT);
```

Obr. 1 - Florian Cramer: *Self.pl*, 2001. Zdroj: <<http://v2.nl/events/executable-code-and-its-cultural-imagination>>.

Toto dielo je programom a básňou zároveň. Program vytvorí prázdny súbor s názvom "self". "Self.pl" číta "self", zatvorí ho a opäť otvorí, aby napísal, čo si prečítal. Algoritmus tohto programu má v angličtine nasledovné znenie: „open it self while it push at it close it open it self print it join at it close it.“<sup>45</sup> Cramer vytvára dielo, ktoré sa nachádza na pomedzí každodenného jazyka, softwaru a poézie.

---

<sup>44</sup>„any programming language is organized by its formal grammar. The meaning of the formal grammar is a system of functions dependent on the language’s grammatical rules and atomic rules (i.e. concrete expressions: variable names, variable values and so on). If there are ways to make all grammar rules sound, then the sound of each rule corresponds to the meaning of this rule, so the sound of the whole program corresponds to the meaning of the program too.“ Malinovski, Ilija. *LYCAY (Let Your Code pLAY)*. *Runme.org – say it with software art!*. [online] 2006 [cit. 10.prosinec 2019], nestr. Dostupné z <<http://runme.org/project/+letyourcodeplay/>>.

<sup>45</sup> Horst, Philipp. Code Poetry’s Aesthetic WW(WEB). In *Another Language: Poetic Experiments in Britain and North America*. Freitag, Kornelia – Vester, Katharina (eds.). Berlin: LIT Verlag, 2008, s. 105.

Zároveň ale odhaľuje ďalšiu sémantickú úroveň: prvý riadok textu funguje ako „brána“, ktorá je kľúčom k otvoreniu softwarového programu za pomoci štandardnej frázy `!#/usr/bin`, nasledovanej informáciou o použitom jazyku, v tomto prípade Perl.<sup>46</sup>

Kódové diela ukazujú, že digitálna poézia nie je závislá na konkrétnom médiu, pretože na seba môže vziať akúkoľvek mediálnu podobu. Pojmy ako umenie a poézia, ktorá ku svojmu vzniku používa počítačový kód ako materiál, neodkazujú k predstavivosti a kreativite počítača, ale ku kreativite ľudskej. Počítačový kód je materiálom umeleckej tvorby ako každý iný umelecký materiál.<sup>47</sup>

### **Počítačový kód a jeho význam v súčasnej kultúre a umení**

V roku 2003 znelo téma festivalu kultúry, umenia a technológie *Ars Electronica Code - The Language of our Time*. Tématicky sa delil na viacero podkategórií, napríklad „kód ako umenie“ alebo „kód ako život“ a v rámci týchto rôznorodých prístupov sa pýtal po vplyve aký majú digitálne kódy na našu kultúru, umenie a spoločnosť. V strede záujmu stál digitálny kód ako materiál kódového umenia, bol ale zároveň elementom, prostredníctvom ktorého sa festival snažil o artikuláciu kategórie „identity“ tejto umeleckej formy. Snahou organizátorov bolo poukázať na otázky súvisiace s paradigmami budúcnosti, ktoré je možné odvodiť z umeleckých (a vedeckých, technických) počínov na úrovni digitálneho kódu a ich vplyv na programovateľnú budúcnosť, ktorej stojíme tvárou v tvár.<sup>48</sup>

Pytagorovská filozofická škola vytvorila koncepciu, ktorá bola založená na vzťahoch medzi číslami. Vo vzťahu k ich otázke usporiadania vesmíru boli prvými, ktorí vyslovili a systematicky pracovali na myšlienke, že symbolický matematický zdrojový kód je základom vesmíru a popisuje zároveň prírodu a kultúru.<sup>49</sup> Ako jeden z piatich princípov nových médií definoval Lev Manovich princíp „prekódovania“.<sup>50</sup> Používa ho k definícii dôsledkov komputelizácie médií, teda procesu, ktorý prevádza médiá na počítačové dáta. Nové médiá chápe na dvoch úrovniach, ako kultúrne (príbeh, komédia, tragédia, kompozícia a podobne) a počítačové (balíčky dát prenášané počítačovou sieťou, počítačový kód a podobne). Z dôvodu, že sú nové médiá vytvárané, ukladané, či archivované za pomoci počítača, Manovich upozorňuje na vpisovanie logiky počítačov hlboko do kultúrnej úrovne médií.<sup>51</sup> Dochádza k vzájomnému ovplyvňovaniu počítačovej a kultúrnej úrovne, keď kultúrna rovina nových

---

<sup>46</sup> Ibid.

<sup>47</sup> Cramer, Florian. *When writing executes itself. Poetry and Program Code*. [online] 2003 [cit. 10. prosince 2019]. Dostupné z <[http://cramer.pleintekst.nl/all/wenn\\_schrift\\_sich\\_selbst\\_ausf%FCht/writing\\_executing\\_itself.rtf](http://cramer.pleintekst.nl/all/wenn_schrift_sich_selbst_ausf%FCht/writing_executing_itself.rtf)>.

<sup>48</sup> Ars Electronica. *CODE - the Language of our Time*. [online] 2003 [cit. 10. prosince 2019]. Dostupné z <[http://90.146.8.18/en/archives/festival\\_archive/festival\\_overview.asp?iPresentationYearFrom=2003](http://90.146.8.18/en/archives/festival_archive/festival_overview.asp?iPresentationYearFrom=2003)>.

<sup>49</sup> Cramer, Florian. *Words Made Flesh: Code, Culture, Imagination*. Rotterdam: Willem de Kooning Academy Hogeschool, 2005.

<sup>50</sup> Manovich, Lev. *Jazyk nových médií*. Praha: Karolinum, 2018.

<sup>51</sup> Ibid., s. 84.

médií, jej obsahy a organizácia je neustále ovplyvňovaná spôsobmi ako počítače formujú náš svet, rozhranie človek – počítač alebo sprístupňujú dáta. Pojem „prekódovanie“ ako princíp nových médií znamená prevedenie do iného formátu. Komputerizácia preniká do hlbších vrstiev našej kultúry a vytvára nové pojmy, náš kultúrny jazyk sa mení pôsobením týchto zmien, sprostredkovaných novými médiami.<sup>52</sup>

Software v kultúrnom kontexte môžeme vnímať ako nástroj ku kreatívnej a kritickej artikulácii spoločenského diania a digitálnych médií, ktoré nás vymedzujú. Podobne ako iné umelecké formy, ani softwarové umenie nemôže byť redukované na technické kritériá, pretože kód je viac, než len nástrojom k uvádzaniu počítača do pohybu. Od iných foriem umenia sa ale môže líšiť tým, že softwaroví umelci komunikujú s médiom počítača pomocou slovných pokynov (logické zápisy) a tak vyzývajú stroj k spolupráci na výslednom diele. Počítačový kód predstavuje v softwarovom umení umelecký konštrukt, ktorý nie je nástrojom k tvorbe estetiky, ale má túto kvalitu sám o sebe. Prostredníctvom rôznych kreatívnych prístupov sa v rámci kultúry realizuje na mnohých úrovniach. Chápeme ho preto z pohľadu existencie jeho estetické hodnoty, v rámci procesu vzniku softwarového umeleckého diela ako aj v momente, keď už existuje ako hotový produkt.

Kultúra sa prostredníctvom digitálnych médií prepisuje do umenia, čítame ju ako umelecký proces, v rámci ktorého sa umelci môžu realizovať a participovať na ňom. Ladislav Kesner si vo vzťahu vizuálnej komunikácie a inštitúcie múzea všíma fenoménu spojeného s nástupom digitálnych technológií a ich reflexie v umeleckej tvorbe.<sup>53</sup> Upozorňuje, že tieto nové obrazové médiá premieňajú spôsoby vnímania vizuálnej kultúry a môžu predstavovať aj zmeny v štruktúre ľudského vnímania. Tam, kde sa spracovávanie informácií stáva kľúčovým, upadá kultúra smerom k jednoduchším výkladom dát a ich významom, k hodnote, ktorú utvárajú a kultúra sa posúva k inému chápaniu samej seba.

## Literatúra

Arns, Inke. Read\_me, run\_me, execute\_me. Code as Executable Text: Software Art and its Focus on Program Code as Performative Text. In *Medien Kunst Netz 2: Thematische Schwerpunkte*. Frieling, Rudolf – Daniels, Dieter (eds.). Wien - New York: Springer, 2005.

Ars Electronica. CODE - the Language of our Time. [online] 2003 [cit. 10.prosinec 2019]. Dostupné z <[http://90.146.8.18/en/archives/festival\\_archive/festival\\_overview.asp?iPresentationYearFrom=2003](http://90.146.8.18/en/archives/festival_archive/festival_overview.asp?iPresentationYearFrom=2003)>.

Cramer, Florian - Gabriel, Ulrike. Software Art. [online] 2001 [cit. 10.prosinec 2019]. Dostupné z <[http://www.netzliteratur.net/cramer/software\\_art\\_-\\_transmediale.html](http://www.netzliteratur.net/cramer/software_art_-_transmediale.html)>.

---

<sup>52</sup> Ibid.

<sup>53</sup> Kesner, Ladislav. *Muzeum umění v digitální době. Vnímání obrazů a prožitky umění v soudobé společnosti*. Praha: ARGO a NG, 2000.

- Cramer, Florian. *Digital Code and Literary Text*. [online] 2001 [cit. 10.prosince 2019]. Dostupné z <[http://www.netzliteratur.net/cramer/digital\\_code\\_and\\_literary\\_text.html](http://www.netzliteratur.net/cramer/digital_code_and_literary_text.html)>.
- Cramer, Florian. *When writing executes itself. Poetry and Program Code*. [online] 2003 [cit. 10.prosince 2019]. Dostupné z <[http://cramer.pleintekst.nl/all/wenn\\_schrift\\_sich\\_selbst\\_ausf%FCHrt/writing\\_executing\\_itself.rtf](http://cramer.pleintekst.nl/all/wenn_schrift_sich_selbst_ausf%FCHrt/writing_executing_itself.rtf)>.
- Cramer, Florian. *Words Made Flesh: Code, Culture, Imagination*. Rotterdam: Willem de Kooning Academy Hogeschool, 2005.
- Cramer, Florian. Language. In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008.
- Eco, Umberto. *O zrcadlech a jiné eseje: znak, reprezentace, iluze, obraz*. Praha: Mladá fronta, 2002.
- Eco, Umberto. *Teorie sémiotiky*. Brno: Janáčkova akademie múzických umění v Brně, 2004.
- Holdcroft, David. *Saussure: signs, system, and arbitrariness*. New York: Cambridge University Press, 1991.
- Horst, Philipp. Code Poetry's Aesthetic WW(WEB). In *Another Language: Poetic Experiments in Britain and North America*. Freitag, Kornelia – Vester, Katharina (eds.). Berlin: LIT Verlag, 2008.
- Kesner, Ladislav. *Muzeum umění v digitální době. Vnímání obrazů a prožitek umění v soudobé společnosti*. Praha: ARGO a NG, 2000.
- Kittler, Friedrich. There is no software. *CTheory.net*. [online] 18. října 1995. Dostupné z <<http://www.ctheory.net/articles.aspx?id=74>>.
- Kittler, Friedrich. Code (or, How You Can Write Something Differently). In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008.
- Krysa, Joasia - Sedek, Grzesiek. Source code. In *Software Studies: A Lexicon*. Fuller, Matthew (ed.). Cambridge, MA: MIT Press, 2008.
- Malinowski, Ilia. LYCAY. *Runme.org – say it with software art!*. [online] 2006 [cit. 10.prosince 2019]. Dostupné z <<http://runme.org/project/+letyourcodeplay/>>.
- Manovich, Lev. *Jazyk nových médií*. Praha: Karolinum, 2018.
- Mukařovský, Jan. Umění jako sémiologický fakt (1934). In *Studie I*. Mukařovský, Jan. Brno: Host, 2000.
- Wall, Larry. *Perl, the first postmodern computer language*. [online] 1999 [cit. 10.prosince 2019]. Dostupné z <<https://cseweb.ucsd.edu/~goguen/courses/230/pm-perl.html>>.
- Wardrip-Fruin, Noah. *Expressive Processing. Digital Fiction, Computer Games, and Software Studies*. Cambridge – London: MIT Press, 2009.