

**TLÜ Haapsalu Kolledž**

**Haridustehnoloogia**

# **Tarkvara**

**Wikipedia artikkel**

**Koostaja: Janika Pannel**

**Haapsalu 2015**

## Sisukord

1	Tarkvara.....	3
2	Tarkvara liigid .....	5
2.1	Programmeerimisvahendid.....	6
3	Tarkvaralised teemad.....	7
3.1	Arhitektuur .....	7
3.2	Teostus .....	7
3.3	Kvaliteet ja töökindlus .....	8
3.4	Litsents .....	8
3.5	Patendid.....	8
4	Väljatöötamine ja rakendamine .....	9
5	Viited .....	10

# 1 Tarkvara

**Tarkvara** ehk **arvuti tarkvara** on masinloetav juhend, mis suunavad arvuti protsessorit spetsiifilisi toiminguid sooritama. Tarkvara hõlmab kõiki mittefüüsilisi arvuti tööks vajalikke või rakenduslikke komponente, eelkõige arvutiprogramme ning nende andmeid – andmefaile, seadeid, dokumentatsiooni, jne.

Tarkvara mõistet kasutatakse vastandina arvuti riistvara mõistele, need on arvuti füüsilised seadmed ja ühendused (näiteks protsessor), millel on tarkvara talletatud ja mis teostavad antud juhised.<sup>1 2</sup>Arvuti riist- ja tarkvara vajavad üksteist ja kumbagi ei saa realselt kasutada ilma teiseta.

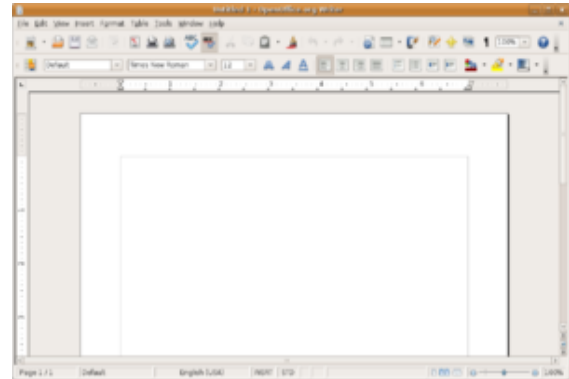


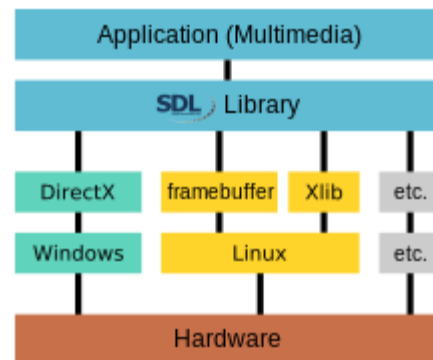
Foto 1 OpenOffice.org Writer

"Tarkvara" on üldine termin. See võib viidata kõikidele arvuti juhistele üldiselt või mistahes konkreetsele juhisele. See hõlmab nii masinjuhiseid (kahendkood, mida protsessor "mõistab") kui ka lähtekoodist (inimesele arusaadavamad juhendid, mis tuleb enne käivitamist tõlkida masinkoodi).

Arvuti tarkvara on komplekt programme, protseduure, funktsioone, seotud andmeid ja nende dokumentatsioone, kui võimalik. Programmi tarkvara täidab rakendatava programmi ülesande, kas andes juhised otse digitaalsele elektroonikale või töötavad sisendina mõne teise tarkvara jaoks.

Erinevalt riistvarast on käivitav tarkvara salvestatud andmetena arvuti süsteemis ning seega ei saa seda füüsiliselt puudutada. Madalaimal tasandil käivitav kood koosneb masinkeelsetest juhistest, mis on spetsiifilised ühele individuaalsele protsessorile. Masinkeel koosneb kahendväärtuste rühmadest, mis tähistavad protsessori juhiseid, mis muudavad arvuti olekut.<sup>3</sup> Näiteks võib mõni juhiseid muuta konkreetseks kohta salvestatud väärtuse (mida kasutaja silmaga ei näe) või (kaudselt) panna arvuti ekraanile ilmuma mõne objekti (mida näeb kasutaja ka ise). Protsessor täidab juhiseid nende esitatud järjekorras, välja arvatud olukorras, kui tal on kästus "hüpata" mõnele teisele käsule või kui toiming on katkestatud.

Tabel 1 Kihstruktuur, mis näitab, kus asub operatsioonisüsteem ja rakendused (rakendusprogrammid).



<sup>1</sup> "Software". computerhope.com Vaadatud 7.12.2013

<sup>2</sup> "Hardware vs Software". diffen.com Vaadatud 7.12.2013

<sup>3</sup> Machine code (machine language)". Margaret Rouse (September 2005). Vaadatud 7.12.2013

Tarkvara on tavaliselt kirjutatud kõrgetasemelises programmeerimiskeeles, mida on inimestel lihtsam ja tõhusam kasutada (sarnaneb rohkem loomuliku keelega) kui masinkeelt. Kõrgetasemelised keeled on koostatud või tõlgitakse masinkeele objektikoodi. Tarkvara võib olla kirjutatud ka assemblerkeeles, sisuliselt meeldejäävam variant masinkeelest, kasutades loomuliku keele tähestikku. Assemblerkeel tuleb koostada objektikoodi koostaja (assembler) abil.

## 2 Tarkvara liigid

Tarkvara hõlmab suurt hulka tooteid, mis võivad olla toodetud erinevaid tehnikaid nagu tavalisi programmeerimiskeeli, mikrokoodi või FPGA konfiguratsiooni kasutades.

Peaaegu kõikidel arvuti platvormidel saab tarkvara jagada paari suuremasse kategooriasse:

- Süsteemitarkvara on tarkvara, mis on kavandatud arvuti riistvara juhendamiseks, et tagada põhifunktsioonid ja tagada platvorm jooksvatele rakendustarkvaradele. Süsteemi tarkvara hõlmab seadme süsteemidraivereid, operatsioonisüsteeme (Mac OS X, Microsoft Windows, Linux jne), serveritarkvara, aknahaldustarkvara ja utiliite (nt diagnostikaprogrammid).
- Süsteemi tarkvara sisaldab ka alglaadimise püsivara, mis laadib lahti (või mõnel juhul ka moodustab) operatsioonisüsteemi. Püsivara on tarkvara, mis on püsivalt salvestatud riistvarasse (täpsemalt püsimällu). Seega on sellel nii tarkvara kui ka riistvara omadusi, kuid on siiski tarkvara.
- Rakendustarkvara ja skriptid olid ajalooliselt määratletud kui kogu tarkvara, mis kasutab arvutisüsteemi, et teha kasulikumat tööd (või lahutada meelt), kui arvuti algoperatsioonid või arvuti ise. Praktikas on aga süsteemi- ja rakendustarkvara erinevused sageli ähmased kasulikkude operatsioonisüsteemi rakenduste tõttu. Rakendustarkvara võimaldab kasutajal teatava kindla ülesande täitmist. Rakendustarkvara alla kuuluvad näiteks kontoritarkvara (Microsoft Office, OpenOffice.org), arhiveerimistarkvara, majandustarkvara, andmebaasid, arvutimängud. Rakendustarkvaras on tavaliselt kasutusel graafiline kasutajaliides (GUI).
- Rakendustarkvara sisaldab töölauarakendusi nagu brauserid ja Microsoft Office, samuti nutitelefonid ja tableti rakendusi (nn "äpp").
- JavaScripti skriptid on tarkvara osad, mis on traditsiooniliselt veebilehtede osad, mis käivitatakse otse brauseri sees, kui veebileht lahti laaditakse, ilma et oleks vaja veebibrauseri pistikprogrammi (inglise *plug-in*). Tarkvara, mis on kirjutatud teistes programmeerimiskeeltes, saab ka veebibrauseris käivitada, kui tarkvara on, kas tõlgitud JavaScripti või kui brauseri pistikprogramm, mis toetab vastavat keelt, on installeeritud. Kõige levinum näide viimasest on ActionScripti skriptid, mida toetab Adobe Flash pistikprogramm.
- Veebirakendused töötavad tavaliselt veebiserveris ja väljundavad veebibrauseritele dünaamiliselt genereeritud veebilehti, kasutades tarkvara nagu PHP, Java või ASP.NET, või isegi JavaScripti, mis jookseb serveris.
- Pistikprogrammid ja laiendid on tarkvara, mis laiendab või muudab mõne teise tarkvara funktsionaalsust, nõudes vastava tarkvara kasutamist, et funktsioneerida.
- Sisseehitatud tarkvara on arvuti tarkvara, mis on kirjutatud, et juhtida masinaid või seadmeid, millest tavaliselt ei mõelda kui arvutitest. Üldiselt spetsialiseerunud eelkõige riistvarale, mida vastav tarkvara jooksub ning millel on aja ja mälu piirangud. Seda terminit kasutatakse mõnikord püsivarana, kuigi püsivara saab kasutada ka arvuti püsimälupõhises koodis, millelt operatsioonisüsteem jookseb, samas kui sisseehitatud tarkvara on tavaliselt ainuke tarkvara kõnealusel seadmel.<sup>4</sup>

---

<sup>4</sup> "Embedded Software—Technologies and Trends". Ebert, Cristof (Mai 2009). Vaadatud 7.12.2013

- Mikrokode on eriline, suhteliselt segast tüüpi sisseehitatud tarkvara, mis ütleb protsessorile, kuidas täita masinkoodi, nii et tegelikult on see madalam tasand kui masinkood.<sup>5</sup> See on tavaliselt salastatud protsessori tootja poolt ning kõik vajalikud mikrokode tarkvara uuendused varustatakse kasutajatele nende poolt (mis on palju odavam kui asendada protsessori riistvara kohaletoimetamine). Seega tavaline programmeerija ei peaks kunagi eeldama, et peab sellega tegelema.

## 2.1 Programmeerimisvahendid

Programmeerimisvahendid on tarkvara programmide või rakenduste kujul, mida tarkvara arendajad (tuntud ka kui programmeerijad, koodijad, häkkerid või tarkvara insenerid) kasutavad, et luua, siluda, säilitada (st parandada või täiustada) või muul viisil toetada tarkvara. Tarkvara kirjutatakse ühes või mitmes programmeerimiskeeles; on olemas palju erinevaid programmeerimiskeeli ja igäühel neist on vähemalt üks rakendus, millest igäüks koosneb oma programmeerimisvahendite komplektist. Need vahendid võivad olla suhteliselt iseseisvad programmid nagu koostajad, silujad, tõlgid ja teksti toimetajad, mida saab kasutada koos, et täita teatud ülesanne (täpselt nagu füüsilise objekti parandamisel mitme tööriista kasutamine). Nad võivad olla ka integreeritud arenduskeskkond (IDE), millel on mitmed või kõik selliseid kompaksete tööriistade funktsioonid. Peaaegu kõik programmeerimiskeele rakendused (erandiks on Smalltalk) annavad võimaluse kasutada konkreetseid vahendeid IDE asemel, sest mõned programmeerijad eelistavad IDE mitte kasutamist erinevatel põhjustel. Lisaks läheb tavaliselt kauem aega, et arendada IDE "vastuvõetavale" tasemele.



<sup>5</sup> "Microcode". Princeton University. Vaadatud 7.12.2013

## 3 Tarkvaralised teemad

### 3.1 Arhitektuur

Kasutajad näevad asju sageli teistmoodi kui programmeerijad. Inimesed, kes kasutavad kaasaegseid üldotstarbelisi arvuteid (erinevalt manussüsteemidest, analoogarvutitest ja superarvutitest) näevad tavaliselt kolme kihti tarkvara, mis täidavad erinevaid ülesandeid: platvormi, rakenduslik ja kasutaja tarkvara.

- Tarkvaraplatvorm: sisaldab püsivara, seadme draiverid, operatsioonisüsteemi ja tavaliselt graafilist kasutajaliidest, mis kokkuvõttes võimaldab kasutajal suhelda arvuti ja selle välisseadmetega. Tarkvaraplatvorm on sageli arvutiga komplekteeritud. Eraarvutitel on kasutajal tavaliselt võimalus muuta tarkvaraplatvormi.
- Rakendustarkvara: rakendustarkvara või rakendused on need, mida enamik inimesi peavad tarkvaraks. Tüüpilised näited on kontoritöö programmid ja videomängud. Rakendustarkvara ostetakse sageli riistvarast eraldi. Mõnikord on rakendused arvutiga kaasas, kuid see ei muuda asjaolu, et nad töötavad iseseisvate rakendustena. Rakendused on tavaliselt sõltumatud programmid operatsioonisüsteemis, kuigi nad on sageli kohandatud teatud platvormidele.
- Kasutaja kirjutatud tarkvara, mida lõppkasutajad ise arendavad ja kujundavad vastavalt endi vajadustele. Kasutaja tarkvara hulka kuuluvad arvutustabeli- ja tekstitöötlusmallid. Isegi e-posti filtrid on omamoodi kasutaja tarkvara. Kasutajad loovad selle tarkvara ise ja tihti ei pane isegi tähele, kui oluline see on. Sõltuvalt sellest, kui asjatundlikult kasutaja kirjutatud tarkvara on integreeritud vaikerakenduse pakettidesse, ei pruugi paljud kasutajad teha vahet originaalpakendil ning töökaaslaste lisatud tarkvaral.

### 3.2 Teostus

Arvuti tarkvara peab laadima arvuti "hoiusesse" (näiteks kõvaketas või mälu). Pärast tarkvara laadimist, on arvuti võimeline tarkvara käivitama. See sisaldab rakendustarkvara juhiste läbimist läbi süsteemi tarkvara riistvarale, mis lõpuks saab juhendid kui masinkoodis. Iga juhise põhjustab arvutis teatud operatsiooni teostamise – andmete liigutamise, arvutuse tegemise või muuta juhiste kontrollvoolu.

Andmete liigutamine tähendab tavaliselt andmete liigutamist ühest mälu kohast teise. Vahel sisaldab see andmete liigutamist mälu ja registreeritud vahel, mis võimaldavad protsessoris andmetele kiire juurdepääsu. Andmete liigutamine, eriti suurtes kogustes, võib olla ohtlik. Seega aeg-ajalt välditakse seda, kasutades andmete viitu. Arvutused sisaldavad lihtsaid toiminguid, näiteks muutuva andme osale väärtuse juurde lisamine. Keerulisemad arvutused võivad sisaldada nii mitmeid operatsioone kui ka andmete elemente.

### 3.3 Kvaliteet ja töökindlus

Tarkvara kvaliteet on väga oluline, eriti äri- ja süsteemitarkvara (näiteks Microsoft Office, Microsoft Windows ja Linux) kvaliteet. Kui tarkvara on vigane, võib see kustutada isiku töö, arvuti kokku jooksutada ja teha muid ootamatuid asju. Paljud vead avastatakse ja kõrvaldatakse läbi tarkvara testimise. Kuid tarkvara testimine eemaldab harva – kui üldse – kõik vead. Mõned programmeerijad isegi ütlevad, et "igal programm on vähemalt üks viga" (Lubarsky seadus). Kõikidel suurtel tarkvarafirmadel, näiteks Microsoftil, Novellil ja Sun Microsystemsil, on oma tarkvara testimise osakonnad, mille konkreetne eesmärk ongi vaid tarkvara testimine. Tarkvara saab kontrollida ühiktestimise, regressioonitestimise ja muude meetodite rakendamisel, mida tehakse käsitsi. Tavaliselt tehakse seda aga automaatselt, sest tihti on testitava koodi suurus üsna suur. Näiteks NASA-l on äärmiselt ranged tarkvara testimise protseduurid paljude operatsioonisüsteemide ja kommunikatsiooni funktsioonide jaoks. Paljud NASA-l põhinevad operatsioonid suhtlevad ja tuvastavad üksteist läbi käskprogrammide, mida kutsutakse käsktarkvaraks. See võimaldab paljudel NASA-s töötaval inimestel kontrollida ja hinnata üldiseid funktsionaalseid süsteeme. Programmid sisaldavad käsk tarkvara võimaldab riistvara insenerja süsteemi tegevuse toimida palju lihtsam koos. Programmide abil, mis sisaldavad käsktarkvara, suudavad riistvara ja tarkvara palju sujuvamalt koos töötada.

### 3.4 Litsents<sup>6</sup>

Tarkvaralitsents annab kasutajale õiguse kasutada tarkvara litsentsitud keskkonnas. Mõnel tarkvaral on litsents, kui see ostetakse n-õ riulilt, riistvaraga kaasas aga OEM litsents. Ülejäänud tarkvaraga kaasneb tasuta tarkvara litsents, andes saajale õigused muuta ja levitada antud tarkvara. Tarkvara võib olla ka vabavara või jaosvara kujul.

### 3.5 Patendid

Tarkvara saab patenteerida mõningates, kuid mitte kõikides riikides. Tarkvarapatendid võivad tarkvara tööstuses olla vastuolulised – paljudel inimestel on selles osas erinevad vaated. Lahkarvamused tarkvarapatentide kohta on seotud konkreetsete algoritmide ja tehnikate kasutamisega, mida tarkvara sisaldab ning mida ei tohi teised dubleerida. Selle dubleerimist peetakse intellektuaalomandi ja autoriõiguse rikkumiseks, sõltuvalt rikkumise raskusastmest.

---

<sup>6</sup> "Software". sezendia.org (2006) Vaadatud 7.12.2013



## 4 Väljatöötamine ja rakendamine

Tarkvara väljatöötamine ja rakendamine sõltub suuresti tarkvara keerukusest. Näiteks Microsoft Wordi tarkvara väljatöötamine ja loomine võtab programmide funktsionaalsete erinevuste tõttu palju rohkem aega kui Microsoft Notepadi arendamine ja väljatöötamine.<sup>7</sup>

Tarkvara on tavaliselt välja töötatud ja loodud (kodeeritud/kirjutatud/programmeeritud) integreeritud arenduskeskkondades (IDE) nagu Eclipse, Emacs ja Microsoft Visual Studio, mis suudavad protsessi lihtsustada ja koostada programmi. Nagu erinevates sektsioonides märgitud, loodakse tarkvara tavaliselt olemasoleva tarkvara ja rakenduse programmeerimisliidese (API), mille antud alumine tarkvara annab nagu GTK +, JavaBeans või Swing, peale. Rakenduse programmeerimisliidese raamatukogud on liigitatud erinevatel põhjustel. Näiteks JavaBeansi teeki (inglise *library*) kasutatakse ettevõtete rakenduste väljatöötamiseks, Windows Formsi teeki aga graafilise kasutajaliidese (GUI) rakenduste väljatöötamisel (näiteks Microsoft Word) ning Windows Communication Foundationi teeki veebiteenuste väljatöötamisel. Programmeerimise alusmõisted quicksort, paisktabel (inglise *hash table*), rida (inglise *array*) ja kahendpuu võivad olla tarkvara loomisel kasulikud. Programmi disainimisel tugineb see rakenduse programmiliidesele.



---

<sup>7</sup> "Design Patterns: Elements of Reusable Object-Oriented Software". Erich Gamma, Ralph Johnson, John Vlissides, Richard Helm. Addison-Wesley Professional 1994

## 5 Viited

1. "Design Patterns: Elements of Reusable Object-Oriented Software". Erich Gamma, Ralph Johnson, John Vlissides, Richard Helm. Addison-Wesley Professional 1994
2. "Embedded Software—Technologies and Trends". Ebert, Cristof (Mai 2009). Vaadatud 7.12.2013
3. "Hardware vs Software". diffen.com Vaadatud 7.12.2013
4. "Machine code (machine language)". Margaret Rouse (September 2005). Vaadatud 7.12.2013
5. "Microcode". Princeton University. Vaadatud 7.12.2013
6. "Software". computerhope.com Vaadatud 7.12.2013
7. "Software". sezindia.org (2006) Vaadatud 7.12.2013