A blue and black logo

Description automatically generatedA blue and white logo

Description automatically generated

Tara Šiler, Blaž Pogelšek

**Kalkulator**

1. seminarska naloga

Maribor, januar 2024

**KAZALO**

[1. UVOD 1](#_Toc157109601)

[2. RAČUNANJE 2](#_Toc157109602)

[3. PRETVARANJE MED ŠTEVILSKIMI SITEMI 3](#_Toc157109603)

[4. LOGIČNA VRATA 4](#_Toc157109604)

[5. PSEVDOKODA 5](#_Toc157109605)

[6. DIAGRAM POTEKA 7](#_Toc157109606)

[VIRI 8](#_Toc157109607)

**KAZALO SLIK**

[Slika 1: Diagram poteka 5](#_Toc157110000)

# 

# UVOD

Pri izdelavi seminarske naloge sva se odločila za uporabo JavaScripta, saj ponuja obsežne možnosti za oblikovanje grafičnega vmesnika. Poleg JavaScripta sva uporabila tudi HTML in CSS, s katerima sva oblikovala izgled kalkulatorja. Za boljšo vizualno prezentacijo sva vključila tudi Bootstrap, ki nudi širok izbor različnih funkcionalnosti in orodij za oblikovanje.

V HTML sva naredila gumbe za vse znake in operatorje, ter vsakemu dodala dogodek (»onclick«), ki sproži izvajanje določene funkcije v JavaScriptu. Poleg tega pa sva oblikovala tudi polje za prikaz vnosov in izpisov ter ustvarila spremenljivko vnos, ki je sprva prazna, ter se nato posodablja ob vnašanju znakov.

Informacije za izdelavo seminarske nalog sva pridobila s pomočjo spletnih strani in YouTube posnetkov navedenih v virih. Prav tako pa nama je pomagalo skozi leto pridobljeno znanje o pretvarjanju med številskimi sistemi in logičnih operatorjih.

# RAČUNANJE

V delu za računanje sva naredila funkcije za dodajanje števil, črk in osnovnih računskih operacij k vnosu ter funkcijo namenjeno izračunavanju vnosa. Ustvarila sva tudi funkcijo, ki počisti celotno vsebino iz polja namenjenega prikazu, ter funkcijo (»brisi«), ki odstrani zadnji element vnosa.

V funkciji oklepaji sva določila kdaj računska operacija zahteva oklepaj in kdaj zaklepaj, in sicer v primeru, da oklepaja še ni ali pa če se je izvedel set oklepaj zaklepaj se napiše oklepaj; zaklepaj pa se vnese v primeru, če imamo oklepaj in ne zaklepaja ali pa ko imamo oklepaj in zaklepaj ter se je pred tem nazadnje pojavil oklepaj. V ločeni funkciji sva za decimalno vejico določila, da se lahko uporabi le ena zapored v posameznem številu.

Za računanje modula sva uporabila operator '%', ki vrne ostanek pri deljenju, za potenciranje operator '\*\*' ter za korenjenje metodo Math.sqrt() [1], [2].

# PRETVARANJE MED ŠTEVILSKIMI SITEMI

Pri tej točki sva omogočila pretvarjanje števil med številskimi sistemi. Začela sva s pretvorbami iz desetiškega številskega sistema v binarni, osmiški ter šestnajstiški. Pretvorbe izvedemo na način, da število delimo z 2/8/16, toliko časa dokler je to število večje od nič, kar nama je omogočila uporaba while zanke. Na koncu pa, s pomočjo for zanke, vrednost preberemo iz desne proti levi.

Nato sva izvedla še pretvorbe iz binarnega, osmiškega in šestnajstiškega številskega sistema v desetiški. Prvi dve pretvorbi sva izvedla na način, da sva, s pomočjo while zanke, najprej dobila zadnjo števko podanega števila ter prvotno število brez te števke. Nato sva zadnjo števko pomnožila z močjo na 2 oz. 8, pri čemer se je eksponent pri vsakem sprehodu skozi zanko povečal za ena (na začetku je eksponent nastavljen na nič). Vsako dobljeno vrednost zmnožka pa sva prištela k iskanem decimalnem številu.

Pri pretvarjanju v šestnajstiški sistem ter iz njega pa sva poleg števil upoštevala tudi črke ter zanje določene vrednosti [3], [4], [5], [6].

# LOGIČNA VRATA

Logična vrata so sestavljena iz osmih funkcij, ki jih lahko razdelimo na tri dele:

1. **Pregled vnosa**:  
- Ta del koda je odgovoren za preverjanje, ali je v vnosu že prisotna katera od logičnih operacij, kot so AND, OR, XOR, NOT, NAND ali NOR.

2. **Delovanje in izpis računanje z logičnimi vrati**:

- V tem delu so definirane funkcije, ki izvajajo specifične logične operacije. Te funkcije obdelujejo vnos in na podlagi izbrane logične operacije izvedejo ustrezno obdelavo binarnih vrednosti [7], [8], [9].

3. **Branje datoteke**:

- Ta del omogoča branje vsebine iz tekstovne datoteke, ki jo uporabnik naloži. Po branju datoteke se vsebina prenese v spremenljivko vnos, ki se nato uporabi za izvajanje logičnih operacij.

# PSEVDOKODA

Klikni želeno stran do katere želiš dostopati

Če (računanje)

Začni vnos

Izvajaš (osnovne računske operacije)

Seštevanje, odštevanje, množenje in deljenje

Izvajaš (ostale računske operacije)

Potenciranje, korenjenje in moduli

Če (si se zmotil)

Napačen element vnosa

Opcija briši

Napačen celoten vnos

Opcija počisti

Sicer

Izračunaj

Če (želiš vnesti nov račun)

Počisti

Nov vnos

Sicer

Zaključi računanje

Če (pretvarjanje med številskimi sistemi)

Vnesi prvotno število

Klikneš na gumb za pretvarjanje v želeni številski sistem

Dobiš rezultat v izbranem številskem sistemu

Če (logična vrata kalulator)

Vnesi logične vrednosti (0 ali 1)

Izberi logično operacijo (AND, OR, XOR, NOT, NAND, NOR)

Dodaj .txt datoteko z želenim računom

Preveri, če je logična operacija že prisotna v vnosu

Če ni prisotna

Dodaj logično operacijo v vnos

Preveri veljavnost vnosa

Če je vnos neveljaven

Izpiši napako: "Napačen vnos"

izvedi izbrano logično operacijo

Izpiši rezultat operacije

Če želiš izvesti novo operacijo

Počisti vnos

Začni nov vnos

Sicer

Končaj program

# DIAGRAM POTEKA

Slika, ki vsebuje besede besedilo, posnetek zaslona, diagram, vrstica

Opis je samodejno ustvarjen

Slika : Diagram poteka

# VIRI

[1] “Evaluate an Equation in Javascript, without eval() - Stack Overflow.” Accessed: Jan. 25, 2024. [Online]. Available: https://stackoverflow.com/questions/11422513/evaluate-an-equation-in-javascript-without-eval

[2] “How To Use JavaScript Exponentiation.” Accessed: Jan. 25, 2024. [Online]. Available: https://www.w3schools.com/howto/howto\_js\_exponentiation.asp

[3] “Program for Binary To Decimal Conversion,” GeeksforGeeks. Accessed: Jan. 25, 2024. [Online]. Available: https://www.geeksforgeeks.org/program-binary-decimal-conversion/

[4] “Binary to Decimal Converter | Calculate Binary to Decimal online,” GeeksforGeeks. Accessed: Jan. 25, 2024. [Online]. Available: https://www.geeksforgeeks.org/binary-to-decimal/

[5] “Program for Octal to Decimal Conversion,” GeeksforGeeks. Accessed: Jan. 25, 2024. [Online]. Available: https://www.geeksforgeeks.org/program-octal-decimal-conversion/

[6] Leo, “Build Decimal to Hexadecimal converter without the toString() JavaScript method,” Stack Overflow. Accessed: Jan. 25, 2024. [Online]. Available: https://stackoverflow.com/q/57912250

[7] “Posts containing ‘logical gates,’” Stack Overflow. Accessed: Jan. 25, 2024. [Online]. Available: https://stackoverflow.com/search?q=logical+gates&s=7e32178f-c22f-488d-b720-67d584ddc6ee&s=c9155ee5-d118-41ad-bf3e-a9dbe3e15348

[8] “Logical operators.” Accessed: Jan. 25, 2024. [Online]. Available: https://javascript.info/logical-operators

[9] “Build software better, together,” GitHub. Accessed: Jan. 25, 2024. [Online]. Available: https://github.com