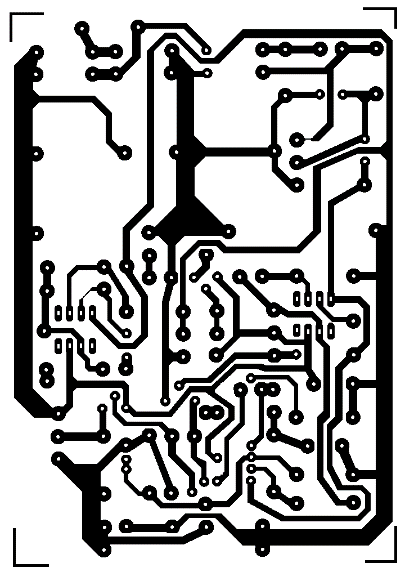
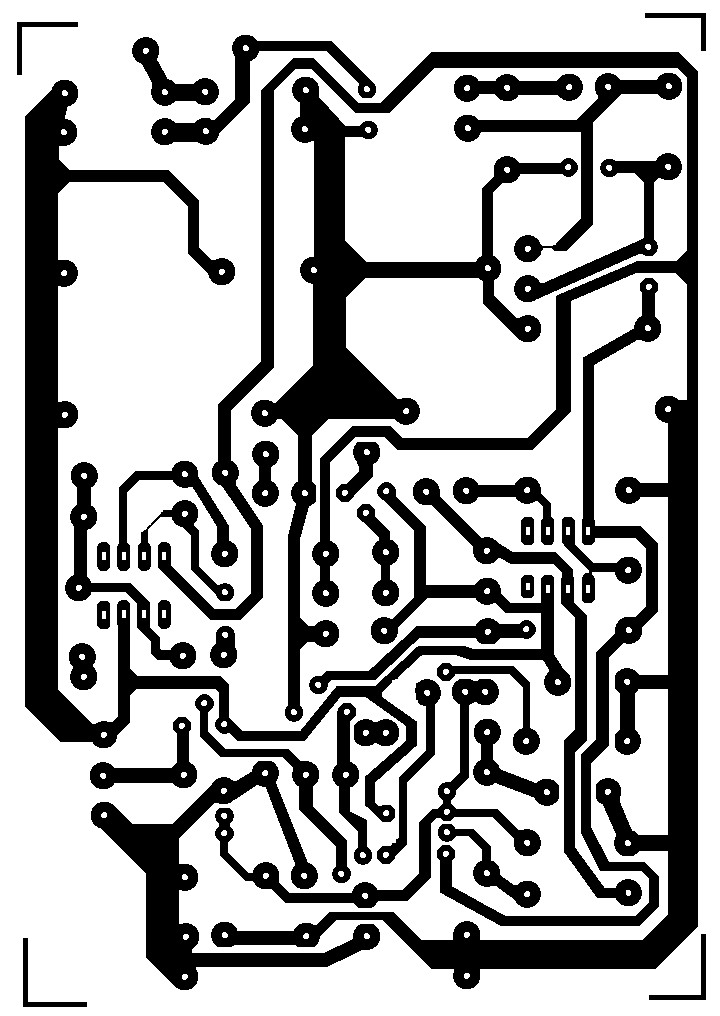
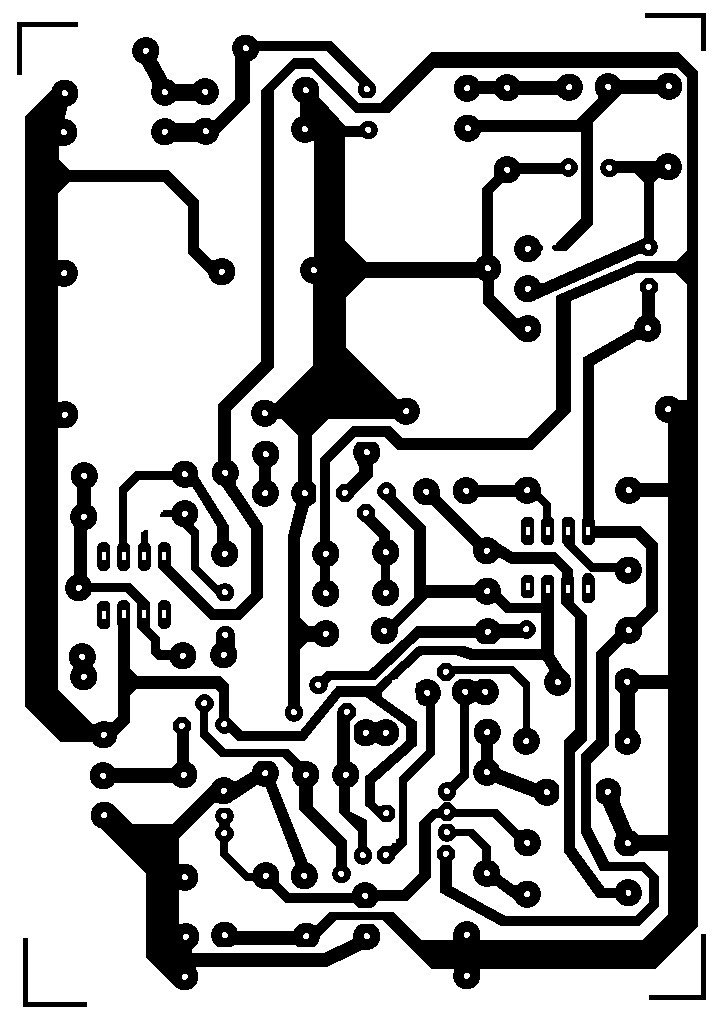
Képi morfológia

Az algoritmusok a main.cpp fájlban találhatóak, 110 – 222 sor, a függvények meghívása a 73-74 sorokban történnek. A feladatban a pcb\_hibas\_8bpp és pcb\_hibas\_8bpp7 képeket használtam. Utóbbi azért, mert a neptun kódomban az első számjegy 8.

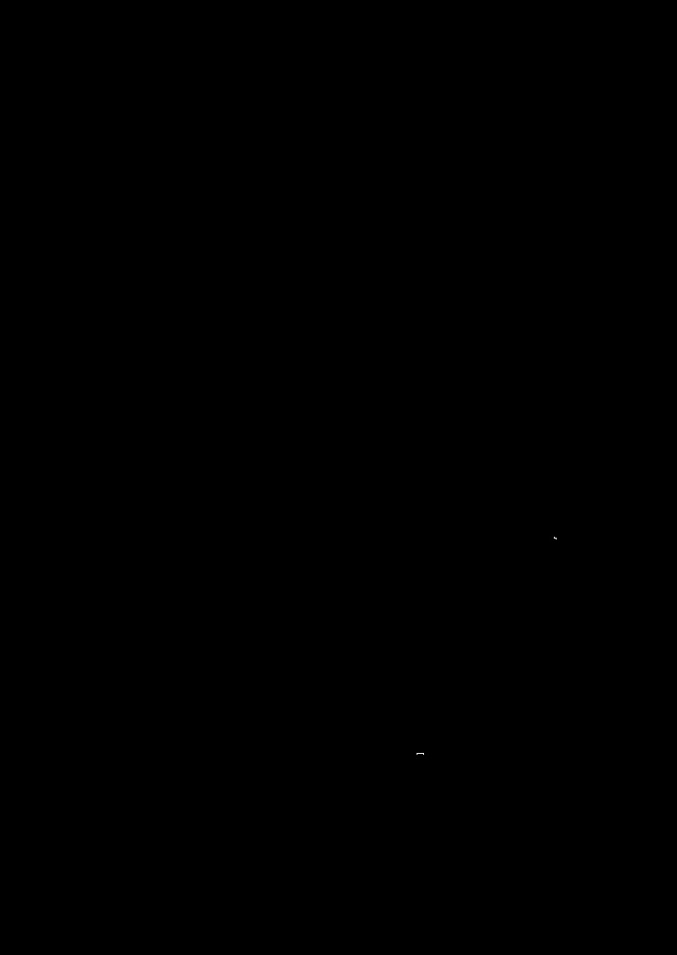
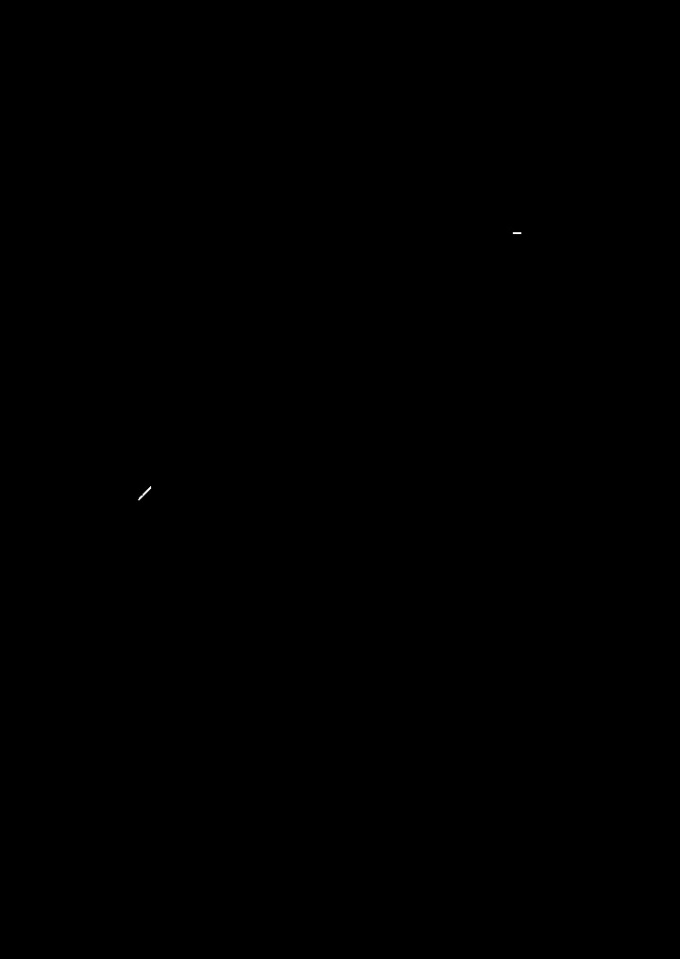
pcb\_hibas\_8bpp eredeti:



pcb\_hibas\_8bpp nyitás után: pcb\_hibas\_8bpp zárás után:



pcb\_hibas\_8bpp túl vékony: pcb\_hibas\_8bpp túl közeli:

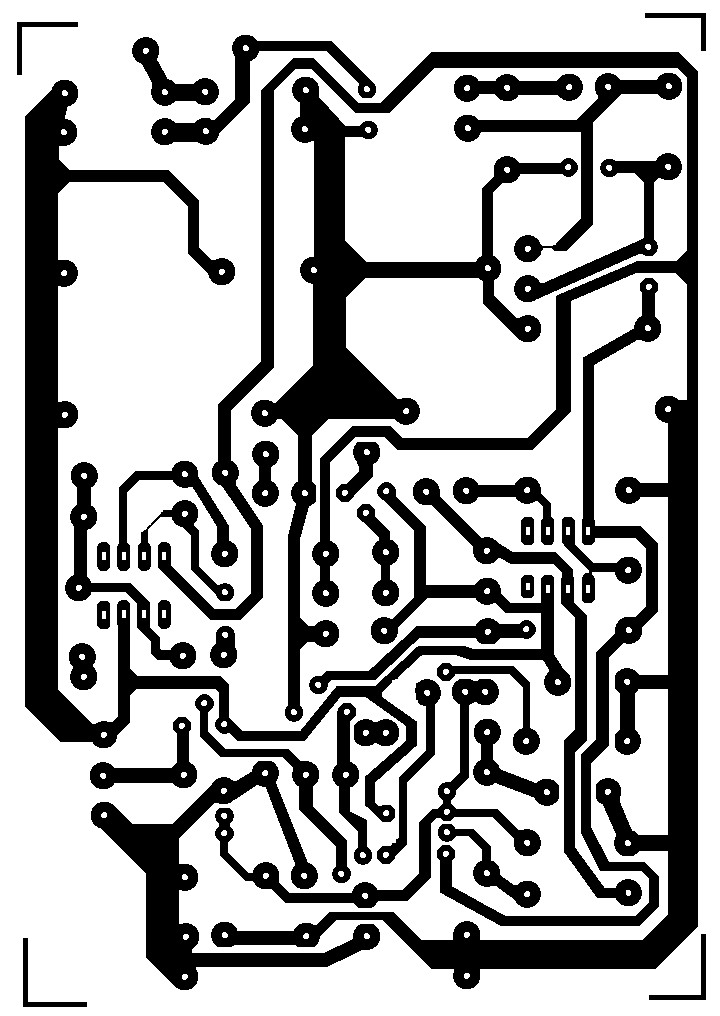
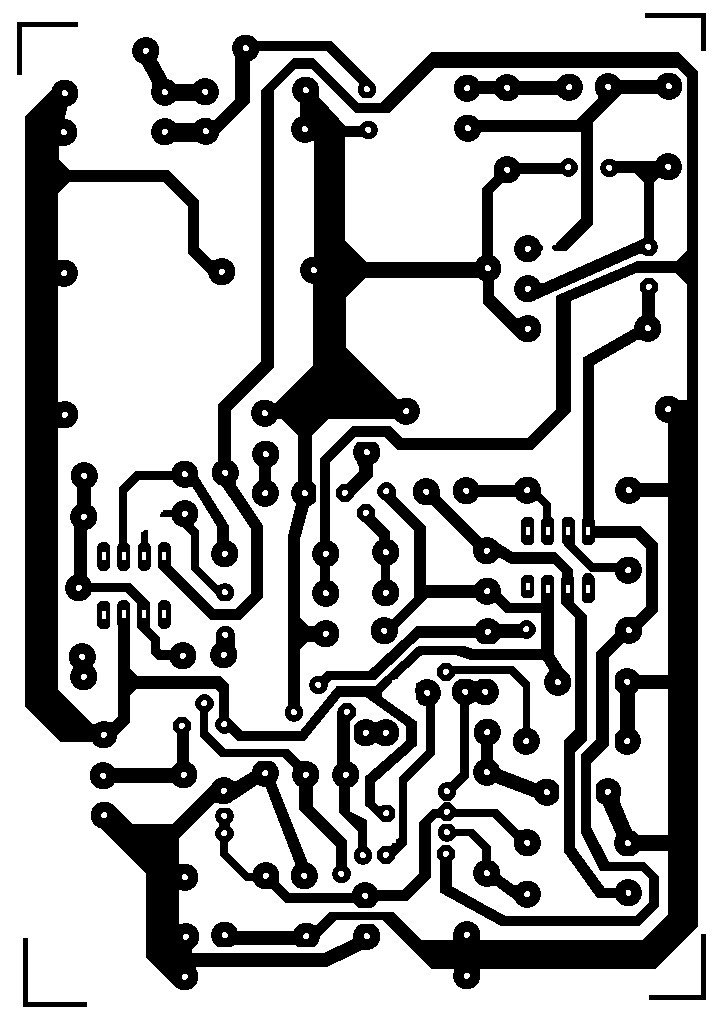


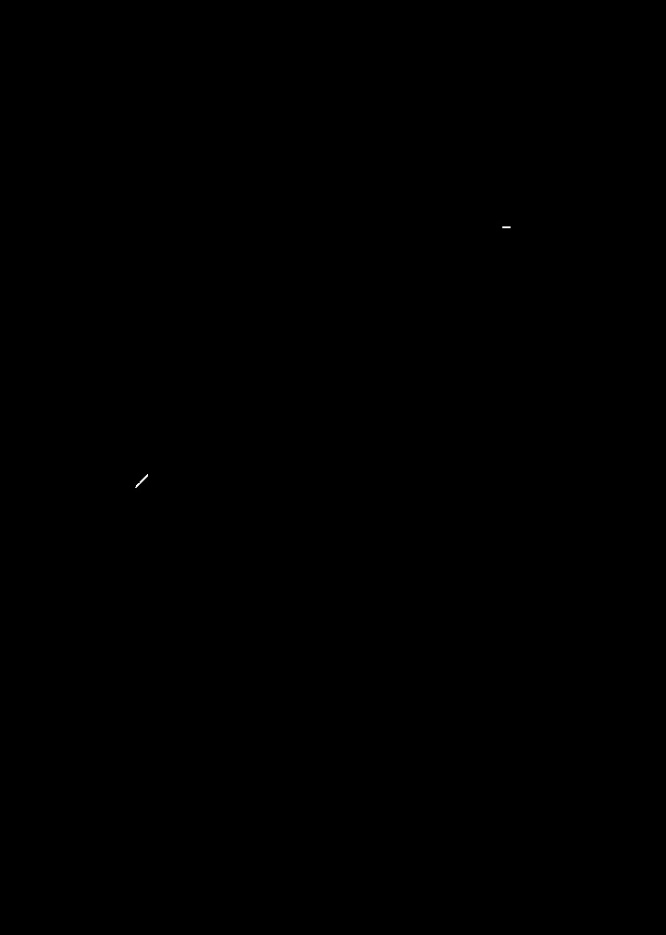
pcb\_hibas\_8bpp7 eredeti

A picture containing icon

Description automatically generated

pcb\_hibas\_8bpp7 nyitás után: pcb\_hibas\_8bpp7 zárás után:



pcb\_hibas\_8bpp7 túl vékony: pcb\_hibas\_8bpp7 túl közeli

Bogárkövető algoritmus

Az algoritmusok a main.cpp fájlban találhatóak, 224 – 400 sor, a függvények meghívása a 76-77 sorokban történnek. A feladatban a bug és bug7 képeket használtam. Utóbbi azért, mert a neptun kódomban az első számjegy 8.

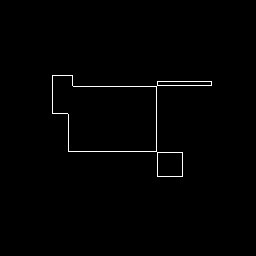
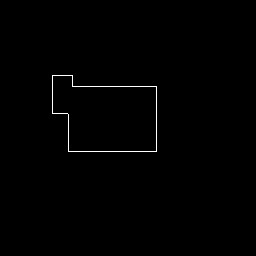
Az algoritmus segítségével körbe járjuk a képeket.

bug:

Logo

Description automatically generated with medium confidence

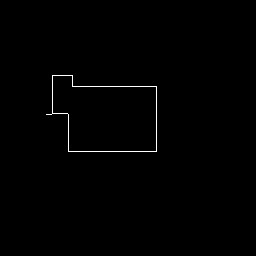
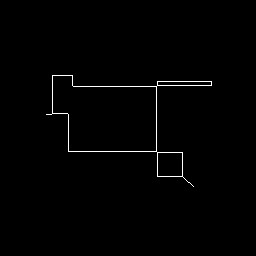
bug bogárkövető algoritmussal: bug visszalépéses bogárkövető algoritmussal:



bug7:

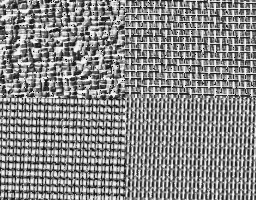
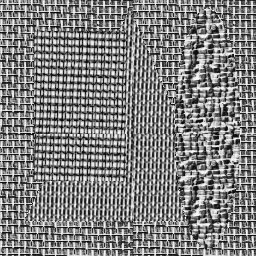
Chart, box and whisker chart

Description automatically generated

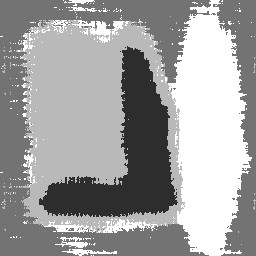
bug7 bogárkövető algoritmussal: bug7 visszalépéses bogárkövető algoritmussal:

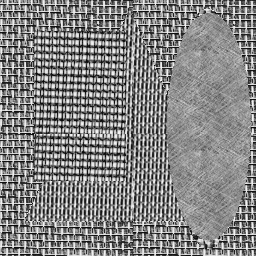
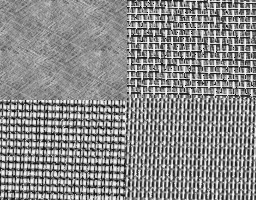
Law szűrő

Az algoritmusok a main.cpp fájlban találhatóak,403 – 565 sor, a függvények meghívása a 79-80 sorokban történnek. A feladatban a laws\_input, laws\_texture és laws\_input5, laws\_texture5 képeket használtam. Utóbbi azért, mert a neptun kódomban az első számjegy 8.

laws\_input: laws\_texture:

laws\_input textúra szűrés után:



laws\_input5: laws\_texture5:

laws\_input5 textúra szűrés után:

