

//Eric Goulart - 2110878

//João Pedro Biscaia Fernandes - 2110361

Questão 4: Faça um programa que leia 2 números reais e imprima o resultado das 4 operações básicas sobre estes 2 números. Verifique o que acontece se o 2º. número da entrada for 0 (zero). Capture o sinal de erro de floating point (SIGFPE) e repita a experiência anterior. Faça o mesmo agora lendo e realizando as operações com inteiros

R: Enquanto trabalhamos com números float o programa não vê problema nenhum em trabalhar com 0, o que acontece é que o resultado da divisão é inf (infinito). Quando mudamos para trabalhar com números inteiros ele captura o sinal de erro, isso porque em Python os inteiros não tem uma representação para o valor infinito.

Questão 5: Faça um programa que tenha um coordenador e dois filhos. Os filhos executam (execvp) um programa que tenha um loop eterno. O pai coordena a execução dos filhos realizando a preempção dos processos, executando um deles por 1 segundo, interrompendo a sua execução e executando o outro por 1 segundo, interrompendo a sua execução e assim sucessivamente. O processo pai fica então coordenando a execução dos filhos, é, na verdade, um escalonador. Faça o processo pai executar por 15 segundos e, ao final, ele mata os processos filhos e termina.

R: O processo pai alterna a execução entre os dois processos filhos usando SIGTOP e SIGCONT. Primeiro criamos os processos filhos usando 'fork', antes de entrar no loop de escalonamento o pai envia um SIGTOP para ambos os filhos, quando entramos no loop de escalonamento que dura 15 segundos o processo pai utiliza de uma flag para ir alternando entre os filhos. Utilizamos SIGTOP quando queremos pausar o filho e SIGCONT quando queremos que ele continue a execução, também utilizamos do sleep(1) para assegurar cada processo rode por 1 segundo. Nossas maiores dificuldades foram em erros bobos de digitação, sem percebê-los ficamos procurando problemas onde não existiam.