

Operációs rendszerek BSc

7.gyak.

2021. 03. 23.

Készítette: Nagy Róbert

Programtervező Informatikus

Neptunkód:JMDRGG

Miskolc, 2021

Minden processz prioritása `p_usrpri = 60`. Round robin nélkül a processzek A, B, C, D sorrendben haladnak. Minden 100-adik clock ticknél ütemezés történik.

[illegible]

...
170	66	24	66	24	64	86	78	16	C	C	
...	
180	66	24	66	24	64	96	78	16	C	C	
...	
190	66	24	66	24	64	106	78	16	C	C	
...	
199	66	24	66	24	64	115	78	16	C	C	KF = 2 / 3
200	64	16	64	16	79	77	90	11	C	A	
201	64	17	64	16	79	77	90	11	A	A	

Minden 100-adik clock tickre van ütemezés és korrekciós faktor. A 100. lépésnél a korrekciós faktor: $KF = 2*2 / (2*2+1)$, A 200. lépésnél pedig : $2*1 / (2*1 + 1)$. A D processz esetén a $p_nice = 5$, azaz egy kicsit alacsonyabb a prioritása mint a többi processzé.

```
int main(int argc, char const *argv[]) {
    int fileDescriptor = open("jmdrgg.txt", O_RDWR);

    if (fileDescriptor == -1) {
        fprintf(stderr, "Hibas file name");
        return 1;
    }

    char text[256];
    int numberOfBytes = read(fileDescriptor, text, sizeof(text));

    if (numberOfBytes == 0 || numberOfBytes == -1) {
        fprintf(stderr, "Hiba volt a beolvasassal");
    }

    text[numberOfBytes] = '\0';

    printf("Beolvasott szoveg: %s\nByteok szama: %d", text, numberOfBytes);

    lseek(fileDescriptor, 0, SEEK_SET);

    char szoveg[] = "szoveg";
    numberOfBytes = write(fileDescriptor, szoveg, sizeof(szoveg));

    if (numberOfBytes == 0 || numberOfBytes == -1) {
        fprintf(stderr, "Hiba volt a fileba irassal");
    }

    printf("\nkiirt szoveg: %s\nByteok szama: %d", szoveg, numberOfBytes);

    close(fileDescriptor);

    return 0;
}
```

A program system callok segítségével beolvas és kiír a megadott filebol.