

# **Operációs rendszerek BSc**

## **Beadandó feladatok**

2021/22/tavaszi\_félév

### **Készítette:**

Berecz Antónia Bsc

Programtervező informatikus

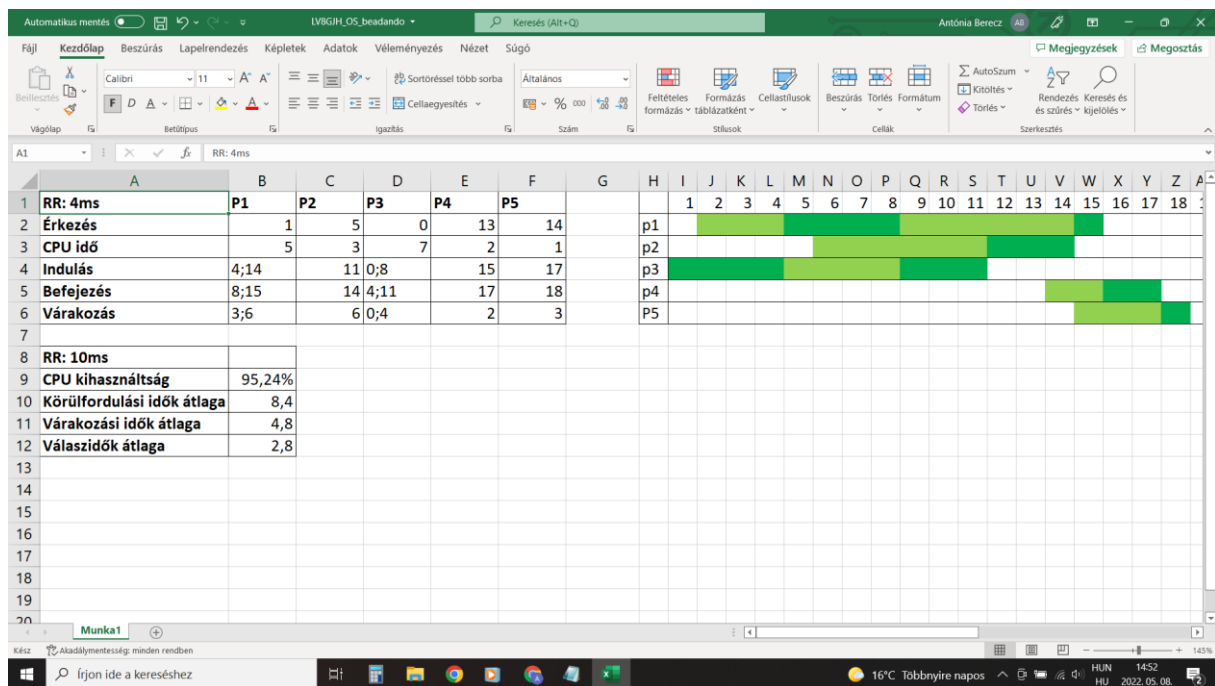
LV8GJH

## 1. feladat:

Adott az alábbi terhelés esetén a rendszer. Határozza meg az indulás, befejezés, várakozás/átlagos várakozás és körülfordulás/átlagos körülfordulás, válasz/átlagos válaszidő és a CPU kihasználtság értékeket az RR: 4 ms ütemezési algoritmusok mellett! (cs: 0,1ms; sch: 0,1ms)

Ábrázolja Gantt diagram segítségével az aktív/várakozó processzek futásának menetét.

Magyarázza a kapott eredményeket!



## Magyarázat:

**CPU kihasználtság:** Ezt a százalékos értéket úgy kaptam meg, hogy az összes CPU időt elosztottam az összes CPU idő és a cs, sch-k összegével.

**Körülfordulási idők átlaga:** Minden processz esetében megnézzük a beérkezési idő és a végleges befejezési idő közötti különbséget, összeadjuk őket, majd elosztjuk az összes processz számával.

**Várakozási idők átlaga:** Összeadjuk az összes várakozási időt, majd elosztjuk az összes processz számával.

**Válaszidők átlaga:** Minden processz esetében megnézzük, hogy mennyi idő telt el a megérkezéstől addig a pillanatig, amíg a processz először megkapta a CPU-t. Ezeket összeadjuk, majd elosztjuk az össze processz számával.

## 2. feladat:

Írjon C nyelvű programot, amely létrehoz egy gyermek processzt, majd küldjön el egy szöveget a szülőtől a gyereknek pipe vezeték segítségével. A gyerek írja ki, hogy hány bájt olvasott, és jelenítse meg az üzenetet szövegét.

```
UW PICO 5.05 File: LV8GJH_QSbeadando.c

#include <stdlib.h>
#include <stdio.h>
#include <sys/wait.h>
#include <sys/types.h>
#include <unistd.h>
#include <string.h>

int main() {
    pid_t pid; //létrehozom a pid változót
    int pipefd[2]; //csővezeték
    if(pipe(pipefd) == -1) //ellenőrzöm, hogy a csővezeték létrejött-e
    {
        perror("pipe error");
        exit(-1);
    }

    char buf;
    char szoveg[32]; //karakterváltozó, amely a beolvasást segíti majd
    //ebben tároljuk majd el a konkrét szöveget

    pid = fork(); //létrehozom a gyermekprocesszt
    if (pid != 0) //szülőben vagyunk
    {
        close(pipefd[0]); //nem fogunk olvasni, így az olvasás fd-t lezárjuk
        strcpy(szoveg, "Milyen szép nyári napunk van.\0");
        write(pipefd[1],szoveg, strlen(szoveg)); //megmondjuk, hogy hová, mit és mennyit írjon
        close(pipefd[1]); //bezárjuk az írás fd-t is
        exit(0); //kilépünk a szülőprocesszból
    }
    else //gyermekben vagyunk
    {
        close(pipefd[1]); //bezárjuk az írás fd-t, mert itt nem fogunk írni
        int db = 0; //létrehozunk egy számlálót

        while((buf = read(pipefd[0], &buf, 1)) > 0)
        {
            db++;
        }

        printf("Ennyi byte-ot olvastam: %d\n", db);
    }
}
```

```
UW PICO 5.05 File: LV8GJH_QSbeadando.c

login as: bereczli
berezli@jerry.iit.uni-miskolc.hu's password:

Welcome to jerry!

berezli@jerry:~$ ./LV8GJH_QSbeadando
berezli@jerry:~$ Milyen szép nyári napunk van.
Ennyi byte-ot olvastam: 31
```