Operációs rendszerek BSc

10. Gyak.

2022. 04. 12.

Készítette:

Csonka Patrik PTI CMU4ZN

1. feladat

33		MAX IGÉN	Y		FOGLAL	3	szabad		MÉG	103
	R1	R2	R3	R1	R2	R3	3,3,2	R1	R2	R
P0	7	5	3	0	1	0		7	4	3
P1	3	2	2	2	0	0	35 B	1	2	2
P2	9	0	2	3	0	2		6	0	0
P3	2	2	2	2	1	1	8	0	1	1
P4	4	3	3	0	0	2		4	3	1
	MANAGÉRIA									
	MAX IGÉN'			FOGLAL			szabad	24	MÉG	
00	R1	R2	R3	R1	R2	R3	5,3,2	R1	R2	R
PO PA	7	5	3	0	1	0	14 14	7	4	3
P1	0	0	0	2	0	0	9	-	-	-
P2	9	0	2	3	0	2		6	0	0
P3	2	2	2	2	1	1	100	0	1	1
P4	4	3	3	0	0	2	12	4	3	1
	MAX IGÉNY				FOGLAL				MÉG	
16	R1	R2	R3	R1	R2	R3	szabad 7,4,3	R1	R2	R
P0	7	5	3	0	1	0		7	4	3
P1	0	0	0	2	0	0	1		(2)	-
P2	9	0	2	3	0	2		6	0	0
P3	0	0	0	2	1	1		-	- 123	. 1
P4	4	3	3	0	0	2		4	3	1
	MAX IGÉNY				FOGLAL			MÉG		
	R1	R2	R3	R1	R2	R3	7,4,5	R1	R2	RS
P0	7	5	3	0	1	0	5, 31	7	4	3
P1	0	0	0	2	0	0	92 9	-	-	-
P2	9	0	2	3	0	2		6	0	0
P3	0	0	0	2	1	1		5	100	-
P4	0	0	0	0	0	2		*	-	-
	MAX IGÉNY			FOGLAL		szabad	MÉG			
	R1	R2	R3	R1	R2	R3	10,4,7	R1	R2	R
PO	7	5	3	0	1	0	1-1-	7	4	3
P1	0	0	0	2	0	0	(A) (A)	-	3 0.50	-
P2	0	0	0	3	0	2	2 2	-	2-3	-
P3	0	0	0	2	1	1	100	-	-	-
P4	0	0	0	0	0	2		2		-
	MAX IGÉNY				FOGLAL			MÉG		
	R1	R2	R3	R1	R2	R3	szabad 10,5,7	R1	R2	R3
PO	0	0	0	0	1	0	,-,-	-	-	-
P1	0	0	0	2	0	0		-	-	
P2	0	0	0	3	0	2	100	-	- 1	-
-	0	0	0	2	1	1	+ +	-	-	3-
		_			_		+			
P3	0	0	0	0	0	2		-	-	-

2. feladat

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main()
    int fd1[2];
   int fd2[2];
   char fixed str[] = " a neptun kodom";
    char input_str[6];
    pid_t p;
    if (pipe(fd1) == -1) {
        fprintf(stderr, "Nem sikerult csovezetek");
        return 1;
    if (pipe(fd2) == -1) {
        fprintf(stderr, "nem sikerult csovezetek");
        return 1;
    scanf("%s", input_str);
    p = fork();
    if (p < 0) {
       fprintf(stderr, "nem sikerult csovezetek");
        return 1;
```

```
else if (p > 0) {
    char concat_str[6];
    close(fd1[0]);
   write(fd1[1], input str, strlen(input str) + 1);
    close (fd1[1]);
   wait (NULL);
   close(fd2[1]);
    read(fd2[0], concat str, 6);
    printf("Vegso sztring: %s\n", concat_str);
   close(fd2[0]);
else {
    close (fd1[1]);
   char concat str[6];
   read(fd1[0], concat str, 6);
    int k = strlen(concat_str);
    int i;
    for (i = 0; i < strlen(fixed str); i++)</pre>
        concat str[k++] = fixed str[i];
   concat str[k] = '\0';
    close(fd1[0]);
    close(fd2[0]);
   write(fd2[1], concat str, strlen(concat str) + 1);
    close (fd2[1]);
   exit(0);
```

Output

```
/tmp/39jflGjExv.o
cmu4zn
Vegso sztring: cmu4zn a neptunkodom
```

3. feladat

```
int main()
   int fd1[2];
   int fd2[2];
   char fixed_str[] = " a nevem";
   char input_str[30];
   pid_t p;
   if (pipe(fd1) == -1) {
       fprintf(stderr, "Nem sikerult csovezetek");
       return 1;
   if (pipe(fd2) == -1) {
       fprintf(stderr, "nem sikerult csovezetek");
   scanf("%s", input_str);
   p = fork();
   if (p < 0) {
       fprintf(stderr, "nem sikerult csovezetek");
       return 1;
```

```
else if (p > 0) {
    char concat str[30];
    close (fd1[0]);
    write(fd1[1], input_str, strlen(input_str) + 1);
    close (fd1[1]);
    wait (NULL);
    close (fd2[1]);
    read(fd2[0], concat_str, 30);
    printf("Vegso sztring: %s\n", concat_str);
    close(fd2[0]);
else {
    close (fd1[1]);
    char concat str[30];
    read(fd1[0], concat_str, 30);
    int k = strlen(concat str);
    int i;
    for (i = 0; i < strlen(fixed str); i++)</pre>
        concat str[k++] = fixed str[i];
    concat str[k] = '\0';
    close(fd1[0]);
    close(fd2[0]);
    write(fd2[1], concat str, strlen(concat str) + 1);
    close(fd2[1]);
    exit(0);
```

```
Output

/tmp/39jflGjExv.o

CsonkaPatrik

Vegso sztring: CsonkaPatrik a nevem
```