

Контрольная работа №2.

(Статья 1, УК ПРАК. За хранение и использование запрещённых материалов, а также списывание, вопрос или ответ соседу по теме или нет – поход на комиссию)

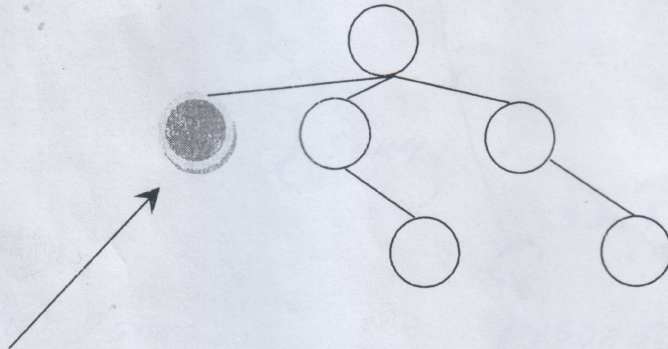
1	2	3	4
+	+	+	+

Вариант 2.

ФИО: Кортов Александр Андреевич

Задание 1. Написать программу.

После запуска программы, она должна породить следующее «генеалогическое» древо процессов:



- после чего этот процесс должен выполнить код из исполнимого файла "/usr/bin/hack".

Замечание: Последовательность рождения процессов и их смерть на Ваше усмотрение.

Задание 2. Написать программу.

На стандартный вход программе подаётся имя текстового файла. Строки указанного текстового файла, содержащие слово «Copyleft», должны выводиться на поток stderr. Все ОСТАЛЬНЫЕ строки вывести на поток stdout.

Замечания: Использовать функции чтения/записи только низкого уровня. Слово «copyleft» считать отличным от «Copyleft».

Задание 3. Написать программу.

На stdin программе подаются имена двух различных текстовых файлов. Ваша программа должна уничтожить содержимое одного из файлов, и скопировать на его место содержимое другого файла.

Замечание: Использовать функции чтения/записи только высокого уровня.

Задание 4. Написать программу.

На stdin программе подаются 10 восьмеричных целых чисел. Реализовать в программе две вспомогательные функции:

- 1) Возвращающую максимальное число из двух переданных в функцию;
- 2) Возвращающую минимальное число из двух переданных в функцию.

Положить указатели на эти функции в массив a_Operations[2]. Подсчитать максимальное и минимальные значения для введённых чисел с помощью a_Operations.

Замечание: Вызов вспомогательных функций напрямую запрещён.

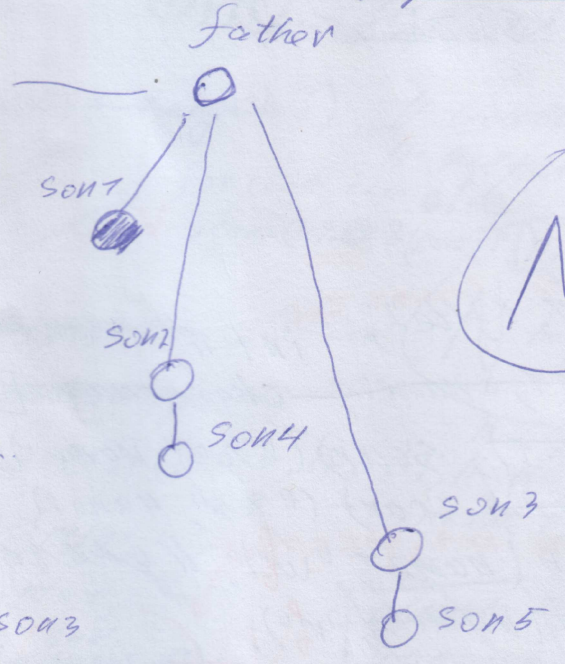
ФИО: Корнев А.А.

№7

```

int main()
{
    int i;
    if (fork())
    { // father / son 1
        if (fork())
        { // father / son 2
            if (fork())
            { // father / son 3
                for (i=0; i<3; i++)
                    wait(NULL);
            }
            else
            { // son 3
                // ...
                exit(0);
            }
        }
        else
        { // son 2
            wait(NULL);
        }
    }
    else
    { // son 1
        // ...
    }
}

```



№7

+

```

if (fork())
{ // son 3 (son 2)
    wait(NULL);
    exit(0);
}
else
{ // son 5 (son 4)
    // ...
    exit(0);
}

```

```

exec("usr/bin/mack", "mack",
NULL);
printf("ERROR\n");
exit(-1);

```


N3

1 2 3 4
v v v v

```
#define SIZE 1024
```

```
int
```

```
int main()
```

```
{
```

```
FILE *file1, *file2; int n;
```

```
char *name1, *name2; char name1[100], name2[100], buf[512];
```

```
scanf(name1); scanf("%s", name1);
```

```
scanf(name2); scanf("%s", name2);
```

```
file1 = fopen(name1, "w"); // you want to write on file
```

```
file2 = fopen(name2, "r");
```

```
while (n = fread(file2, buf, SIZE) != 0)
```

```
{
```

```
    fwrite(file1, buf, n);
```

```
}
```

```
printf("Done!\n");
```

```
return 0;
```

```
}
```

```
fclose(file1);
```

```
fclose(file2);
```

+


```
#define SIZE 2024
#define s_char sizeof(char)
int main()
```

```
{
```

```
char c;
char buf[SIZE];
int fd;
```

N2

```
read_str(buf, fd);
fd = open(buf, O_RDONLY);
```

```
while (read_str(buf, fd) != 0)
```

```
{ if (str_in(buf, "CopyLeft") == 1)
  write_str(buf, stderr);
  else
    write_str(buf, stdout);
```

```
}
close(fd);
return 0;
```

```
}
```

```
int read_str(char *buf, int fd)
```

```
{
  n = 0;
  i = 0;
  char c;
  while ((i = read(fd, buf, s_char)) != 0 && (c != '\n'))
```

```
{
  buf[n] = c;
  n++;
}
```

```
if (i == 1)
  buf[n] = c; // = '\n';
```

```
return n;
```

```
}
```

~~read_str~~
~~write_str~~

read_str
write_str

str_in
mode & permissions
(chmod mode)

~~1) write_str~~
~~2) open~~
~~3) write~~
~~4) write~~
~~5) write~~
~~6) write~~
~~7) write~~
~~8) write~~
~~9) write~~
~~10) write~~
~~11) write~~
~~12) write~~
~~13) write~~
~~14) write~~
~~15) write~~
~~16) write~~
~~17) write~~
~~18) write~~
~~19) write~~
~~20) write~~
~~21) write~~
~~22) write~~
~~23) write~~
~~24) write~~
~~25) write~~
~~26) write~~
~~27) write~~
~~28) write~~
~~29) write~~
~~30) write~~
~~31) write~~
~~32) write~~
~~33) write~~
~~34) write~~
~~35) write~~
~~36) write~~
~~37) write~~
~~38) write~~
~~39) write~~
~~40) write~~
~~41) write~~
~~42) write~~
~~43) write~~
~~44) write~~
~~45) write~~
~~46) write~~
~~47) write~~
~~48) write~~
~~49) write~~
~~50) write~~
~~51) write~~
~~52) write~~
~~53) write~~
~~54) write~~
~~55) write~~
~~56) write~~
~~57) write~~
~~58) write~~
~~59) write~~
~~60) write~~
~~61) write~~
~~62) write~~
~~63) write~~
~~64) write~~
~~65) write~~
~~66) write~~
~~67) write~~
~~68) write~~
~~69) write~~
~~70) write~~
~~71) write~~
~~72) write~~
~~73) write~~
~~74) write~~
~~75) write~~
~~76) write~~
~~77) write~~
~~78) write~~
~~79) write~~
~~80) write~~
~~81) write~~
~~82) write~~
~~83) write~~
~~84) write~~
~~85) write~~
~~86) write~~
~~87) write~~
~~88) write~~
~~89) write~~
~~90) write~~
~~91) write~~
~~92) write~~
~~93) write~~
~~94) write~~
~~95) write~~
~~96) write~~
~~97) write~~
~~98) write~~
~~99) write~~
~~100) write~~

~~open (mode, flags)~~
~~read (fd, buf, count)~~
~~write (fd, buf, count)~~

на гр. компютер
-11- функция

Вариант 2.

ФИО: Корнев А.А

```

int write_str(int buf, int stream)
{
    int i;
    for (i=0; (buf[i]!='\n') && (i < SIZE); i++)
    {
        write(stream, &(buf[i]), 1);
    }
    if (i < SIZE && buf[i] == '\n') // если в
        write(stream, &(buf[i]), 1); // последней строке
    return 0; // abc \n abc \n de
}

```

int stream? Знаю же
 N2
 программа

(-)
 (+)

```

int str_in(int buf, int size, char *str)
{
    int i=0; int n=0;
    while (i < size)
    {
        if (buf[i] == *str) // ("Copy left", ..., "Copy left") ✓
            if (buf[i] != str[n]) // не найдено
                n=0;
            else
                n++;
        if (n == strlen(str))
            return 1;
        i++;
    }
    return 0;
}

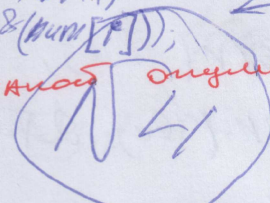
```

как-то
 как...
 (прог :)

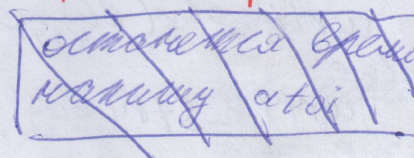

```
int f_min(int a, int b)
int f_max(int a, int b)
```

ФНО: Корнов А.А

```
int main (int argc, char *argv[]) // считаем, что блог
{
    // не переименовать
    int max;
    int min;
    int i;
    int num[10];
    Operations[2];
    // ... 
    for (i=0; i<10; i++)
        scanf ("%d", &(num[i]));
    // sum max &
    // sum min
    // ... 
    int (*a_Operations[2]) (int a, int b); // указатели
    // на функции
    a_Operations[0] = f_min;
    a_Operations[1] = f_max;
    max = min = atoi (num[0]);
    for (i=1; i<10; i++)
    {
        // ... 
        min = (a_Operations[0]) (min, num[i]);
        max = (a_Operations[1]) (max, num[i]);
    }
    printf ("Max value: %d \n Min value: %d \n", max, min);
    return 0;
}
```



читаем жер, а не строка



```
int f_min (int a, int b)
{
    if (a < b)
        return a;
    else
        return b;
}
```

```
int f_max (int a, int b)
{
    if (a > b)
        //
    }
}
```

atoi -
преобраз.
символы
в число
(желам).