

**Министерство образования и науки Российской Федерации
Федеральное агентство по образованию
Государственное образовательное учреждение высшего профессионального
образования
«Московский государственный технический университет имени Н.Э. Баумана»
(МГТУ им. Н.Э. Баумана)
Факультет «Робототехника и комплексная автоматизация» (РК)
Кафедра «Системы автоматизированного проектирования» (РК6)**



**Отчет по лабораторной работе №3 по курсу
«Операционные системы»**

Студент: Сергеева Диана

Группа: РК6-56Б

Преподаватель: Грошев С.В.

Проверил:

Дата:

13 вариант

Задание «Тренажер клавиатуры»:

Требуется разработать программу, которая реализует терминальный интерфейс тренажёра клавиатуры для консольного ввода текстовых строк по образцу. Желаемая длина образцовой строки должна передаваться программе через аргумент командной строки её вызова. Образец строки заданной длины должен формироваться в программе по случайной последовательности, состоящей из отображаемых символов кодовой таблицы ASCII, которую завершает символ перевода строки. Полученный таким способом образец строки должен быть отображён в потоке стандартного вывода. После этого в программе должен быть установлен неканонический режим терминального ввода без эхо-печати сигналы прерывания и завершения. Этот режим должен обеспечивать контролируемый ввод символьной строки, совпадающей с образцом, где входной символ должен приниматься программой и отображаться в потоке стандартного ввода только при его совпадении с образцом. При этом любые ошибки ввода должны блокировать стандартный вывод образца. Стандартный ввод должен продолжаться до получения входной строки, совпадающей с образцом. После завершения ввода по образцу в потоке стандартной диагностики должно отображаться информационное сообщение, которое идентифицирует продолжительность ввода и число её принудительного завершения при нажатии клавиши Escape в любой момент времени до ввода всех символов образца. В любом случае перед завершением программы должен быть восстановлен исходный режим терминального ввода. При разработке программы необходимо использовать системные вызовы read, write, time, sbrk и exit, а также системные функции терминального интерфейса termios OS UNIX и стандартные библиотечные функции системы программирования C для генерации случайных последовательностей.

Разработать программу тренажера клавиатуры с использованием терминального интерфейса, в которой должен вводиться правильный результат арифметической операции сложения двух операндов. Первый операнд вводится вручную, а второй генерируется случайным образом. Следует предусмотреть блокировку ввода ошибочных цифр и других символов.

Исходный код:

```
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <termios.h>
#include <stdio.h>
int textmode(int);
int randline();
```

```

int getch();
static char pattern[80];
static char op1[80];
static char op2[80];
static int res;
int textmode(int mode) {
static struct termios con[2];
if(mode > 0)
return(tcsetattr(0, TCSAFLUSH, &con[1]));
tcgetattr(0, &con[0]);
tcgetattr(0, &con[1]);
con[0].c_lflag &= ~(ICANON | ECHO | ISIG);
con[0].c_iflag &= ~(ISTRIP | IXOFF | IXANY | IXON);
con[0].c_oflag |= CS8;
con[0].c_cc[VMIN] = 2;
con[0].c_cc[VTIME] = 1;
return(tcsetattr(0, TCSAFLUSH, &con[0]));
}
int getch() {
unsigned char c[2];
static int len=0;
/*if(len > 1) {
c[0] = len;
len = 0;
return(c[0]);
}
*/
c[0] = c[1] = 0;
if((len = read(0, c, 2)) < 2)
return(c[0]);
if(c[0] == 27)
c[0] = 0;
/* len = c[1]; */
ungetc(c[1], stdin);
return(c[0]);
}

```

```

}
int randline(int len) {
int i=0;
int r;
/*
srand(getpid());
while(i < len) {
pattern[i++] = rand() % (10) + 48;
}
*/
snprintf(pattern, 80,"%d",atoi(op1)+atoi(op2));
return(0);
}
int main(int argc, char* argv[]) {
int len;
int i=0;
int err=0;
unsigned char c;
len = atoi(argv[1]);
snprintf(op1, 80,"%s",argv[1]);
char b[80];
srand(getpid());
b[0] = rand() % (10) + 48;
snprintf(op2, 80,"%c",b[0]);
randline(len);
/*
write(1, pattern, len);
write(1, "\n", 1);
*/
write(1, op1, strlen(op1));
write(1, "+", 1);
write(1, op2, strlen(op2));
write(1, "=\n", 2);
textmode(0);
while(i < strlen(pattern)) {

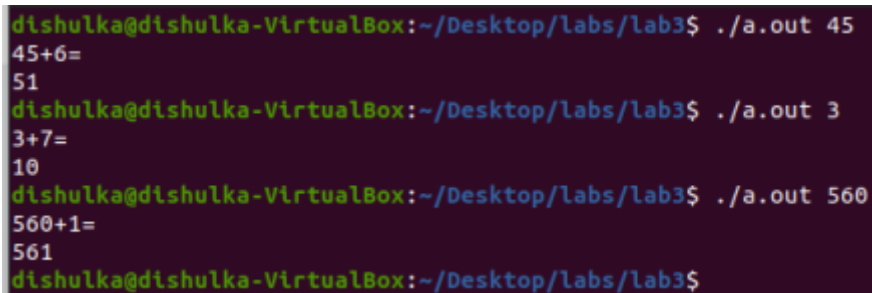
```

```

switch(c = getch()) {
case 0: c = '\007';
switch(getch()) {
case 67: c = pattern[i];
break;
case 68: if(i == 0)
break;
i--;
write(1, "\b", 1);
continue;
default: break;
}
break;
case 27: i = len;
c = '\007';
break;
default: if(c != pattern[i])
c = '\007';
break;
}
(c == '\007') ? err++ : i++;
write(1, &c, 1);
}
write(1, "\n", 1);
textmode(1);
return(err);
}

```

Результат работы программы:



```

dishulka@dishulka-VirtualBox:~/Desktop/labs/lab3$ ./a.out 45
45+6=
51
dishulka@dishulka-VirtualBox:~/Desktop/labs/lab3$ ./a.out 3
3+7=
10
dishulka@dishulka-VirtualBox:~/Desktop/labs/lab3$ ./a.out 560
560+1=
561
dishulka@dishulka-VirtualBox:~/Desktop/labs/lab3$

```