Program

```c
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]){
        if(argc!=3){
                printf("Error");
                return 1;
        }
        int num1 = atoi(argv[1]);
        int num2 = atoi(argv[2]);
        printf("Sum: %d\n", num1+num2);
        return 0;
}
```

Program 2

```c
#include <stdio.h>
#include <unistd.h>
#include <sys/wait.h>

int main(){
        pid_t pid = fork();
        if(pid<0){
                printf("Fork failed");
                return 1;
        }
        else if(pid==0){
                char *argv[] = {"./myadder","10","20",NULL};
        execvp(argv[0], argv);
        perror("execvp failed");
        return 1;
        }
        else{
                wait(NULL);
                printf("Child process has finished\n");
        }
        return 0;
}
```

1 Sample run of the program

```
s23a40@Server-2:~/blab$ nano exp12a.c
s23a40@Server-2:~/blab$ gcc exp12a.c -o myadder
s23a40@Server-2:~/blab$ gcc exp12b.c
s23a40@Server-2:~/blab$ ./a.out
Sum: 30
Child process has finished
s23a40@Server-2:~/blab$
```