ASSIGNMENT 5
10/10/23

NAME : SHRESTH SONKAR
REGNO : 20214272
GROUP : CS5D
TOPIC : OS LAB
CODE : CS-15203

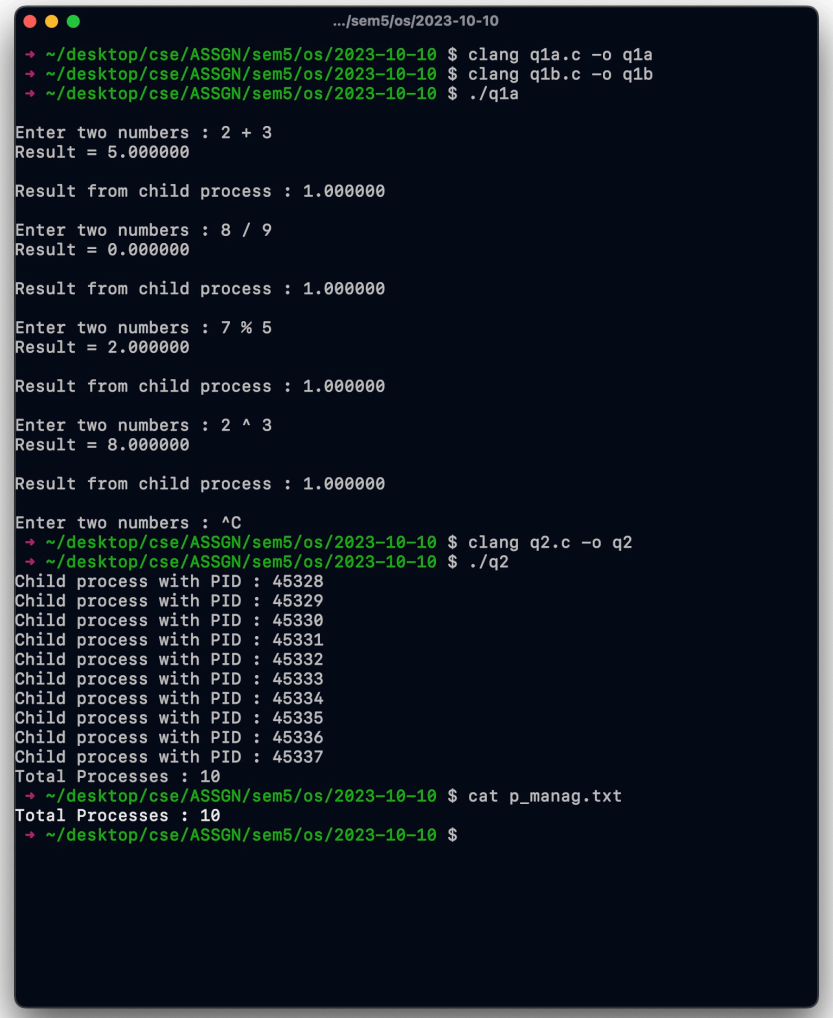
```c
// Q1a
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>
#include <sys/types.h>

int main() {
    int num1, num2;
    char op;
    while (1) {
        printf("\nEnter two numbers : ");
        scanf("%d %c %d", &num1, &op, &num2);
        pid_t childpid = fork();

        if (childpid == -1) {
            perror("fork\n");
            exit(1);
        }

        if (childpid == 0) {
            char num1_str[12], num2_str[12];
            sprintf(num1_str, "%d", num1);
            sprintf(num2_str, "%d", num2);
            char op_str[2] = {op, '\0'};

            execl("./q1b", "./q1b", num1_str, num2_str,
op_str, (char *) NULL);
            perror("execl\n");
            exit(1);
        } else {
            int status;
            waitpid(childpid, &status, 0);
            if (WIFEXITED(status)) {
                double result = WIFEXITED(status);
                printf("\nResult from child process :
%lf\n", result);
            }
        }
    }
    return 0;
}
```

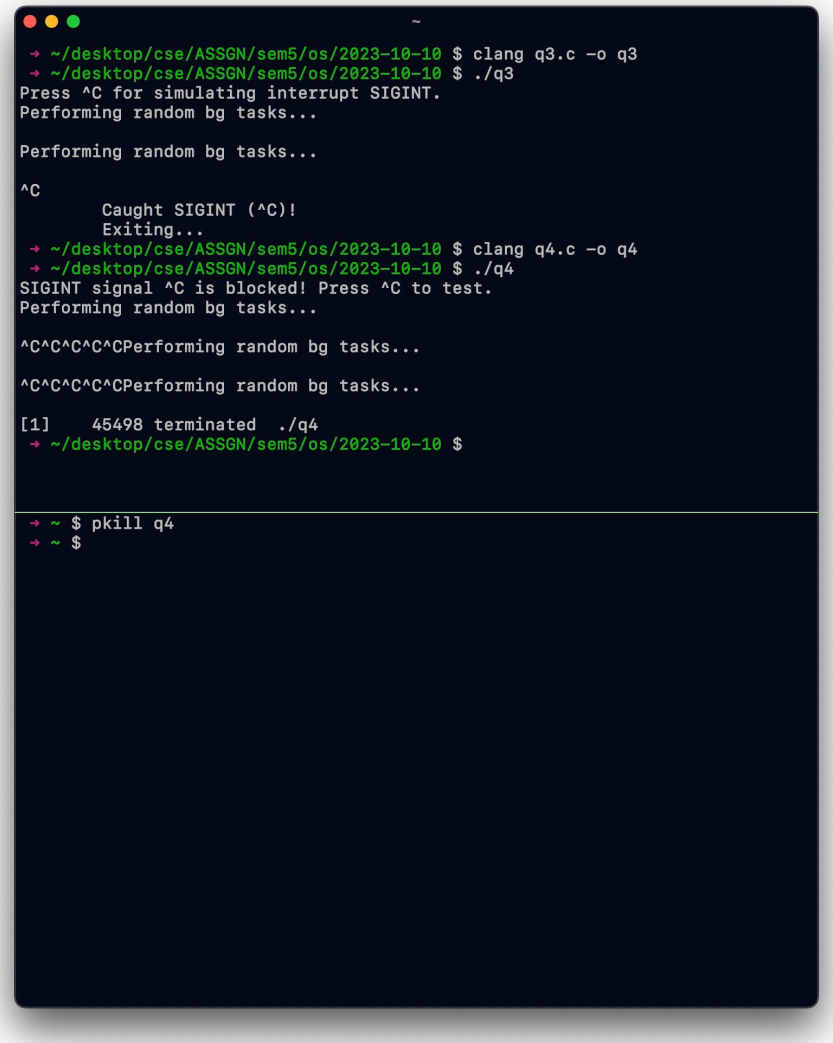
```c
// Q1b
#include <math.h>
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char **argv) {
    int num1 = atoi(argv[1]);
    int num2 = atoi(argv[2]);
    char op = argv[3][0];
    double result;

    if (op == 'e')
        exit(0);
    else if (op == '+')
        result = num1 + num2;
    else if (op == '-')
        result = num1 - num2;
    else if (op == '*')
        result = num1 * num2;
    else if (op == '/')
        result = num1 / num2;
    else if (op == '%')
        result = num1 % num2;
    else if (op == '^')
        result = pow(num1, num2);
    else {
        fprintf(stderr, "INVALID OPERATION %c !\n",
op);
        exit(1);
    }

    printf("Result = %lf\n", result);
    return result;
}
```

```c
// Q2
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>
#include <sys/types.h>

int main() {
    int tot_proc = 0;
    int i;
    for (i = 1; i <= 10; i++) {
        pid_t childpid = fork();

        if (childpid == -1) {
            perror("fork\n");
            exit(1);
        }

        if (childpid == 0) {
            printf("Child process with PID : %d\n",
getpid());
            exit(0);
        } else {
            wait(NULL);
            tot_proc++;
        }
    }

    FILE *file = fopen("p_manag.txt", "w");
    if (file == NULL) {
        perror("fopen\n");
        exit(1);
    }
    fprintf(file, "Total Processes : %d\n", tot_proc);
    fclose(file);

    system("cat p_manag.txt");
    return 0;
}
```

```c
// Q3
#include <stdio.h>
#include <stdlib.h>
#include <signal.h>
#include <unistd.h>

void sigint_hndlr(int signum) {
    printf("\n\tCaught SIGINT (^C)!\n\tExiting...\n");
    sleep(1);
    exit(1);
}

int main() {
    if (signal(SIGINT, sigint_hndlr) == SIG_ERR) {
        perror("signal\n");
        return 1;
    }

    printf("Press ^C for simulating interrupt SIGINT.
\n");
    while (1) {
        printf("Performing random bg tasks...\n\n");
        sleep(2);
    }

    return 0;
}
```

```c
// Q4
#include <stdio.h>
#include <stdlib.h>
#include <signal.h>
#include <unistd.h>

int main() {
    sigset_t blk_msk;
    sigemptyset(&blk_msk);
    sigaddset(&blk_msk, SIGINT);

    if (sigprocmask(SIG_BLOCK, &blk_msk, NULL) == -1) {
        perror("sigprocmask\n");
        return 1;
    }

    printf("SIGINT signal ^C is blocked! Press ^C to
test.\n");
    while (1) {
        printf("Performing random bg tasks...\n\n");
        sleep(2);
    }

    return 0;
}
```


.../sem5/os/2023-10-10
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ clang q1a.c -o q1a
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ clang q1b.c -o q1b
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ ./q1a
Enter two numbers : 2 + 3
Result = 5.000000
Result from child process : 1.000000
Enter two numbers : 8 / 9
Result = 0.000000
Result from child process : 1.000000
Enter two numbers : 7 % 5
Result = 2.000000
Result from child process : 1.000000
Enter two numbers : 2 ^ 3
Result = 8.000000
Result from child process : 1.000000
Enter two numbers : ^C
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ clang q2.c -o q2
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ ./q2
Child process with PID : 45328
Child process with PID : 45329
Child process with PID : 45330
Child process with PID : 45331
Child process with PID : 45332
Child process with PID : 45333
Child process with PID : 45334
Child process with PID : 45335
Child process with PID : 45336
Child process with PID : 45337
Total Processes : 10
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ cat p_manag.txt
Total Processes : 10
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $


~
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ clang q3.c -o q3
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ ./q3
Press ^C for simulating interrupt SIGINT.
Performing random bg tasks...
Performing random bg tasks...
^C
Caught SIGINT (^C)!
Exiting...
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ clang q4.c -o q4
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $ ./q4
SIGINT signal ^C is blocked! Press ^C to test.
Performing random bg tasks...
^C^C^C^C^CPerforming random bg tasks...
^C^C^C^C^CPerforming random bg tasks...
[1] 45498 terminated ./q4
➜ ~/desktop/cse/ASSGN/sem5/os/2023-10-10 $
➜ ~ $ pkill q4
➜ ~ $