

## A)Using Fork

Code:

```
#include <stdio.h>
```

```
#include <unistd.h>
```

```
#include <sys/types.h>
```

```
#include <sys/wait.h>
```

```
void performAddition(int array1[],int array2[],int size){
```

```
    printf("Child process performing task\n");
```

```
    for(int i=0; i<size; i++){
```

```
        int sum= array1[i]+array2[i];
```

```
        printf("Sum of %d +  
%d=%d\n",array1[i],array2[i],sum);
```

```
    }
```

```
}
```

```
void performSubtraction(int array1[],int array2[],int size){
```

```

    printf("parent process performing task\n");
    for(int i=0; i<size; i++){
        int sub= array1[i]-array2[i];
        printf("Subtraction of %d -
%d=%d\n",array1[i],array2[i],sub);
    }
}

```

```

int main() {
    pid_t pid;
    int array1[] = {1, 2, 3, 4, 5};
    int array2[] = {6, 7, 8, 9, 10};
    int size = sizeof(array1) / sizeof(array1[0]);

    pid = fork();

    if (pid == 0) {
        sleep(5);
        performAddition(array1, array2, size);
        printf("Getpid()=%d\n",getpid());
    }
    else if (pid > 0) {

```

```

        wait(NULL);

        performSubtraction(array1, array2, size);

        printf("Getppid()=%d\n",getppid());
    } else {

        printf("Fork failed.\n");

        return 1;

    }

    return 0;
}

```

Output:

```

Activities  Terminal  May 13 14:52
cse@cse: ~/B49

Subtraction of 1 - 6=-5
Subtraction of 2 - 7=-5
Subtraction of 3 - 8=-5
Subtraction of 4 - 9=-5
Subtraction of 5 - 10=-5
cse@cse:~/B49$ gedit practical5.c
^C
cse@cse:~/B49$ gcc practical5.c
cse@cse:~/B49$ ./a.out
Child process performing task
Sum of 1 + 6=7
Sum of 2 + 7=9
Sum of 3 + 8=11
Sum of 4 + 9=13
Sum of 5 + 10=15
parent process performing task
Subtraction of 1 - 6=-5
Subtraction of 2 - 7=-5
Subtraction of 3 - 8=-5
Subtraction of 4 - 9=-5
Subtraction of 5 - 10=-5
cse@cse:~/B49$ gedit practical5.c
^C
cse@cse:~/B49$ gcc practical5A.c
cse@cse:~/B49$ ./a.out
Child process performing addition:
1 + 6 = 7
2 + 7 = 9
3 + 8 = 11
4 + 9 = 13
5 + 10 = 15
Parent process performing subtraction:
1 - 6 = -5
2 - 7 = -5
3 - 8 = -5
4 - 9 = -5
5 - 10 = -5
cse@cse:~/B49$

```

## B) Using vfork

Code:

```
#include <stdio.h>
```

```
#include <unistd.h>
```

```
#include <sys/types.h>
```

```
#include <sys/wait.h>
```

```
void performAddition(int array1[], int array2[], int size) {  
    printf("Child process performing addition:\n");  
    for (int i = 0; i < size; i++) {  
        int result = array1[i] + array2[i];  
        printf("%d + %d = %d\n", array1[i], array2[i], result);  
    }  
}
```

```
void performSubtraction(int array1[], int array2[], int size) {  
    printf("Parent process performing subtraction:\n");  
    for (int i = 0; i < size; i++) {  
        int result = array1[i] - array2[i];  
        printf("%d - %d = %d\n", array1[i], array2[i], result);  
    }  
}
```

```
}
```

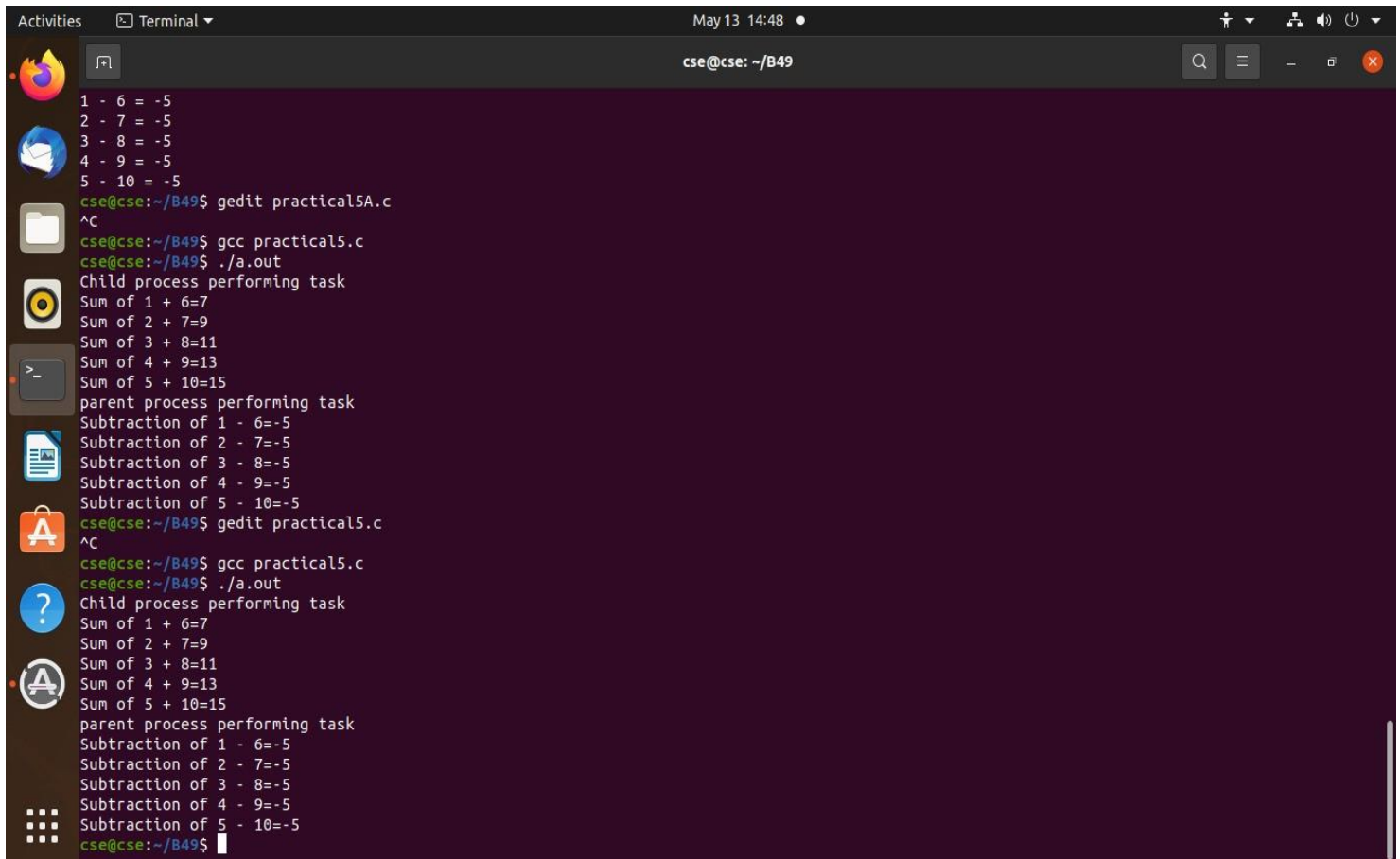
```
int main() {  
    pid_t pid;  
    int array1[] = {1, 2, 3, 4, 5};  
    int array2[] = {6, 7, 8, 9, 10};  
    int size = sizeof(array1) / sizeof(array1[0]);  
  
    pid = vfork();  
  
    if (pid == 0) {  
        // Child process  
        performAddition(array1, array2, size);  
        _exit(0);  
    } else if (pid > 0) {  
        // Parent process  
        wait(NULL);  
        performSubtraction(array1, array2, size);  
  
    } else {  
        // Fork failed  
        printf("Fork failed.\n");  
        return 1;  
    }  
}
```

}

return 0;

}

Output:



```
Activities Terminal May 13 14:48 cse@cse: ~/B49
1 - 6 = -5
2 - 7 = -5
3 - 8 = -5
4 - 9 = -5
5 - 10 = -5
cse@cse:~/B49$ gedit practical5A.c
^C
cse@cse:~/B49$ gcc practical5.c
cse@cse:~/B49$ ./a.out
Child process performing task
Sum of 1 + 6=7
Sum of 2 + 7=9
Sum of 3 + 8=11
Sum of 4 + 9=13
Sum of 5 + 10=15
parent process performing task
Subtraction of 1 - 6=-5
Subtraction of 2 - 7=-5
Subtraction of 3 - 8=-5
Subtraction of 4 - 9=-5
Subtraction of 5 - 10=-5
cse@cse:~/B49$ gedit practical5.c
^C
cse@cse:~/B49$ gcc practical5.c
cse@cse:~/B49$ ./a.out
Child process performing task
Sum of 1 + 6=7
Sum of 2 + 7=9
Sum of 3 + 8=11
Sum of 4 + 9=13
Sum of 5 + 10=15
parent process performing task
Subtraction of 1 - 6=-5
Subtraction of 2 - 7=-5
Subtraction of 3 - 8=-5
Subtraction of 4 - 9=-5
Subtraction of 5 - 10=-5
cse@cse:~/B49$
```