

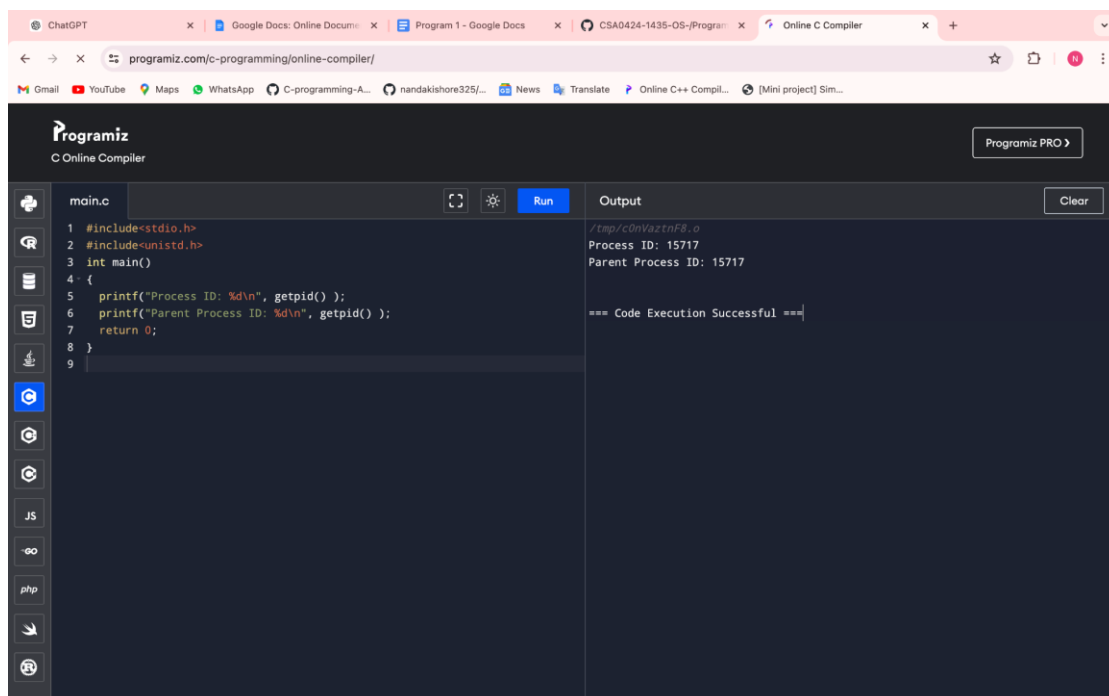
Aim :

Create a new process by invoking the appropriate system call. Get the process identifier of the currently running process and its respective parent using system calls and display the same using a C program.

Algorithm :

- Fork the process using `fork()` system call.
- Check if the fork was successful; if not, print error and exit.
- In the child process, get and print the child's process ID and its parent's process ID.
- In the parent process, get and print the parent's process ID and its parent's process ID.
- Ensure both processes complete properly.
- Exit the program.

Program & Output:



The screenshot shows the Programiz Online C Compiler interface. The code editor on the left contains the following C program:

```
1 #include<stdio.h>
2 #include<unistd.h>
3 int main()
4 {
5     printf("Process ID: %d\n", getpid() );
6     printf("Parent Process ID: %d\n", getppid() );
7     return 0;
8 }
9
```

The output window on the right displays the results of the program execution:

```
/tmp/c0nVaztnF8.o
Process ID: 15717
Parent Process ID: 15717

=== Code Execution Successful ===
```

Conclusion : This program correctly demonstrates how to retrieve and print the process ID and the parent process ID using the `getpid()` and `getppid()` system calls in C.