#include <stdio.h>

#include <stdlib.h>

#include <unistd.h>

#include <sys/wait.h>

// Global variable

int global\_var = 10;

int main()

{

pid\_t pid;

// Create a child process

pid = fork();

if (pid < 0)

{

fprintf(stderr, "Fork failed.\n");

return 1;

}

else if (pid == 0)

{

// This code is executed by the child process

printf("Child process modifying the global variable.\n");

global\_var = 20; // Modify the value of the global variable in the child process

printf("Child process - Global variable value: %d\n", global\_var);

}

else

{

// This code is executed by the parent process

printf("Parent process waiting for child to modify the global variable.\n");

// Wait for the child to finish

wait(NULL);

printf("Parent process - Global variable value: %d\n", global\_var);

}

getch ();

return 0;

}

OUTPUT:

Parent process waiting for child to modify the global variable.

Child process modifying the global variable.

Child process - Global variable value: 20

Parent process - Global variable value: 20