Index

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.No | Contents | Page No. | Date | Remarks | Signature |
| 1 | Predict the Output of the following program | 2 | 07-Feb-2022 |  |  |
| 2 | Calculate number of times hello is printed | 3 |  |
| 3 | Predict the Output of the following program | 4 |  |
| 4 | Predict the Output of the following | 6 |  |
| 5 | Write a Program to find process id and parent process id. | 8 |  |
| 6 | Write a program to fork a child and print the process id of parent and child process | 9 |  |
| 7 | Program to do Sum of numbers from 1 to 10, by dividing the job into two processes (parent and one child) in C Programming. | 11 |  |

Lab Assignment 7

# **Ques1. Predict the Output of the following program**

*// Predict Output of Following Program*

#include <stdio.h>

#include <sys/types.h>

#include <unistd.h>

int main()

{

    fork();

    printf("Hello World");

    return 0;

}

# **Output**

# **Ques 2. Calculate number of times hello is printed**

#include <stdio.h>

#include <sys/types.h>

#include <unistd.h>

int main()

{

    fork();

    fork();

    fork();

    printf("Hello World\n");

    return 0;

}

# **Output**

## **Hello will be printed 8 times**

# **Ques 3. Predict the Output of the following program**

#include <stdio.h>

#include <sys/types.h>

#include <unistd.h>

void forkexample()

{

    if (fork() == 0)

    {

        printf("Hello from Child\n");

    }

    else

    {

        printf("Hello from Parent\n");

    }

}

int main()

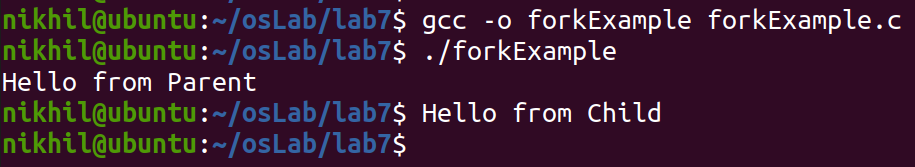
{

    forkexample();

    return 0;

}

# **Output**



# **Ques 4. Predict the Output of the following**

#include <stdio.h>

#include <sys/types.h>

#include <unistd.h>

void forkexample()

{

    int x = 1;

    if (fork() == 0)

        printf("Child has x = %d\n", ++x);

    else

        printf("Parent has x = %d\n", --x);

}

int main()

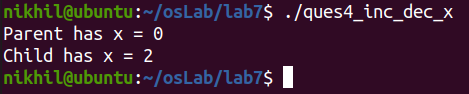
{

    forkexample();

    return 0;

}

# **Output**



# **Ques 5 Write a Program to find process id and parent process id.**

#include <stdio.h>

#include <sys/types.h>

#include <unistd.h>

int main()

{

    printf("hello process is %d of parent :%d\n", getpid(), getppid());

    fork();

    printf("child : %d of parent %d \n", getpid(), getppid());

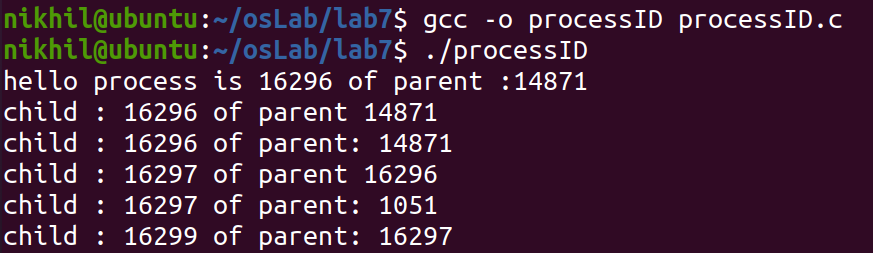
    fork();

    printf("child : %d of parent: %d\n", getpid(), getppid());

    return 0;

}

# **Output**



# **Ques 6. Write a program to fork a child and print the process id of parent and child process**

#include <stdio.h>

#include <sys/types.h>

#include <unistd.h>

int main()

{

    int i;

    printf("hello before fork \n");

    printf("i : %d\n", i);

    i = fork();

    printf("\n");

    if (i == 0)

    {

        printf("Child has started\n\n");

        printf("child printing first time \n");

        printf("getpid : %d getppid : %d \n", getpid(), getppid());

        sleep(5);

        printf("\nchild printing second time \n");

        printf("getpid : %d getppid : %d \n", getpid(), getppid());

    }

    else

    {

        printf("parent has started\n");

        printf("getpid : %d getppid : %d \n", getpid(), getppid());

        printf("\n");

    }

    printf("Hi after fork i : %d\n", i);

    return 0;

}

# **Output**



# **Ques 7. Program to do Sum of numbers from 1 to 10, by dividing the job into two processes (parent and one child) in C Programming.**

#include <stdio.h>

#include <unistd.h>

int main()

{

    int a[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

    int sum1 = 0, sum2 = 0, sum = 0;

    int id;

    id = fork();

    if (id == 0)

    {

        for (int i = 0; i < 10; i++)

        {

            sum1 += a[i];

            sum += sum1;

        }

        printf("Parent Sum: %d\n", sum1);

    }

    else

    {

        for (int i = 1; i < 10; i++)

        {

            sum2 += a[i];

            sum += sum2;

        }

        printf("Child Sum: %d\n", sum2);

    }

    if (id == 0)

    {

        printf("Final Sum is: %d\n", sum);

    }

    return 0;

}

# **Output**

