

OS_LAB_07_Assignment

CE_054

Aim :- Study of Dup2 system call and Excel function.

Theory:

1) dup2 system call:

SYNOPSIS

```
#include <unistd.h>

int dup2(int oldfd, int newfd);
```

DESCRIPTION

This system call creates a copy of the file descriptor oldfd.

dup2() makes newfd be the copy of oldfd, closing newfd first if necessary, but note the following:

- * If oldfd is not a valid file descriptor, then the call fails, and newfd is not closed.
- * If oldfd is a valid file descriptor, and newfd has the same value as oldfd, then

dup2() does nothing, and returns newfd.

After a successful return from this system call, the old and new file descriptors may be used interchangeably. They refer to the same open file description and thus share file offset and file status flags.

The two descriptors do not share file descriptor flags (the close-on-exec flag). The close-on-exec flag for the duplicate descriptor is off.

RETURN VALUE

On success, this system call returns the new descriptor. On error, -1 is returned, and errno is set appropriately.

2) execl function:

SYNOPSIS

```
#include <unistd.h>

extern char **environ;

int execl(const char *path, const char *arg, ...);
```

DESCRIPTION

The const char *arg and subsequent ellipses in the execl(), execlp(), and execl() functions can be thought of as arg0, arg1, ..., argn. Together they describe a list of one or more pointers to null-terminated strings that represent the argument list available to the executed program.

The list of arguments must be terminated by a NULL pointer, and, since these are variadic functions, this pointer must be cast (char *) NULL.

RETURN VALUE

The execl() functions only return if an error has occurred. The return value is -1, and errno is set to indicate the error.

Programs :-

1. Write a program to implement ls | sort functionality using the system calls and functions covered in the lab.

Code:-

```
// Author : Dhruv B kakadiya
#include<stdio.h>
#include<unistd.h>
#include<sys/wait.h>
#include<sys/stat.h>
#include<fcntl.h>
#include<string.h>

void main()
{
    int pid, pret, pipefd[2], status;
    ;
    pret = pipe(pipefd);
    if (pret == -1)
    {
        printf("\n\nCould not create pipe.\n\n");
    }
    else
    {
        {
            pid = fork();
            if (pid == -1)
            {
                printf("\n\nCould not create pipe.\n\n");
            }
            else if (pid > 0)
            {
                wait(&status);
                close(pipefd[1]);
                printf("Entered in Parent process\n\n");
            }
        }
    }
}
```

```

        dup2(pipefd[0], 0);
        execl("/bin/sort", "sort", (char*)0);
    }
    else{
        close(pipefd[0]);
        printf("---Entered in Child process---\n");
        dup2(pipefd[1], 1 );
        execl("/bin/ls", "ls", "-l", (char*)0);
    }
}
}

```

Output :-

```

dhruvkakadiya@kali: ~/Desktop
File Actions Edit View Help
dhruvkakadiya@kali:~/Desktop$ ls -l
total 32
-rwxr-xr-x 1 dhruvkakadiya dhruvkakadiya 16904 Aug 30 09:07 a.out
-rw-rw-rw- 1 dhruvkakadiya dhruvkakadiya 682 Aug 30 09:07 prog1.c
-rw-rw-rw- 1 dhruvkakadiya dhruvkakadiya 1016 Aug 30 09:05 prog2.c
lrwxrwxrwx 1 dhruvkakadiya dhruvkakadiya 73 Jul 6 09:55 start-tor-browser.desktop → /home/dhruvkakadiya/Downloads/tor-browser_en-US/start-tor-browser.desktop
dhruvkakadiya@kali:~/Desktop$ gcc prog1.c
dhruvkakadiya@kali:~/Desktop$ ./a.out
---Entered in Child process---
Entered in Parent process

lrwxrwxrwx 1 dhruvkakadiya dhruvkakadiya 73 Jul 6 09:55 start-tor-browser.desktop → /home/dhruvkakadiya/Downloads/tor-browser_en-US/start-tor-browser.desktop
-rw-rw-rw- 1 dhruvkakadiya dhruvkakadiya 1016 Aug 30 09:05 prog2.c
-rw-rw-rw- 1 dhruvkakadiya dhruvkakadiya 682 Aug 30 09:07 prog1.c
-rwxr-xr-x 1 dhruvkakadiya dhruvkakadiya 16904 Aug 30 09:53 a.out
total 32
dhruvkakadiya@kali:~/Desktop$

```

2. Write a program to achieve following:

- => Child process should open a file with the contents to be sorted, pass the contents to parent process.
- => Parent process should sort the contents of the file and display.

Code :-

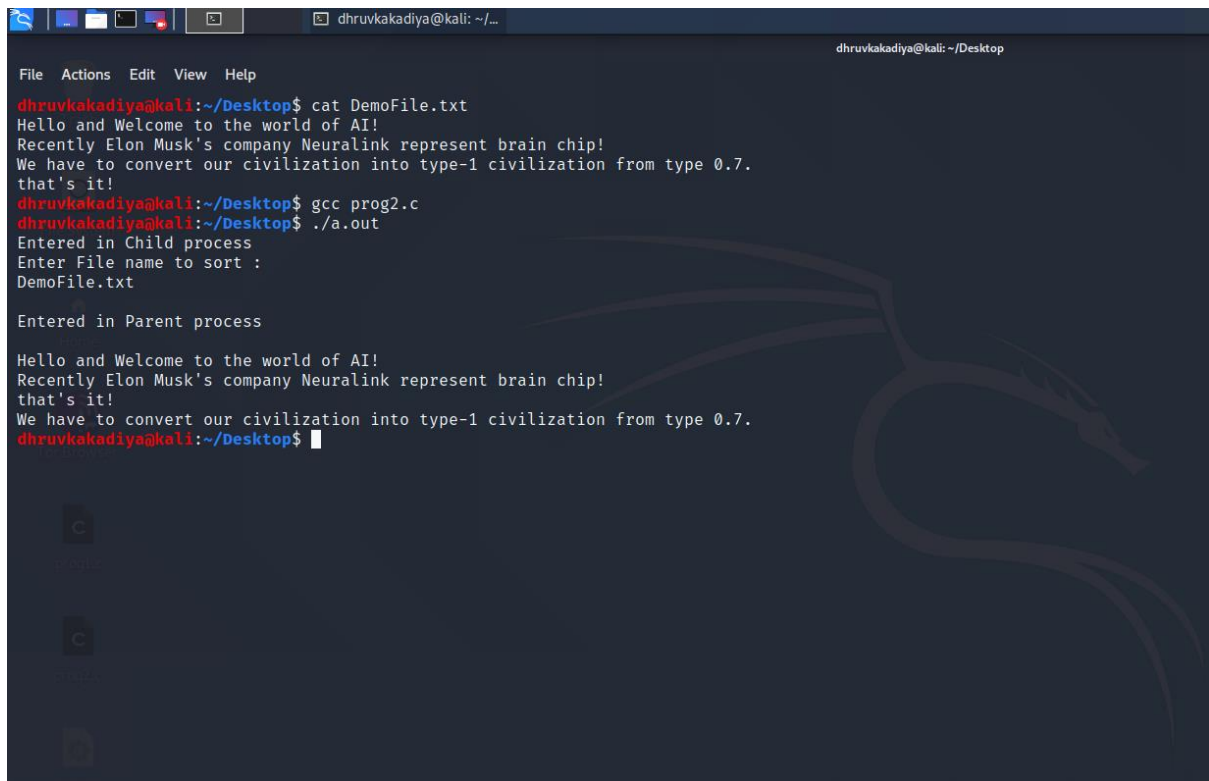
```

// Author : Dhruv B Kakadiya
#include<stdio.h>
#include<unistd.h>
#include<sys/wait.h>
#include<sys/stat.h>
#include<fcntl.h>
#include<string.h>

void main()
{
    int pid, pret, pipefd[2], status, filehand, rd;
    char filedata[1000];
    char filenm[100];
    pret = pipe(pipefd);
    if (pret == -1)
    {
        printf("\n\nCould not create pipe.\n\n");
    }
    else
    {
        {
            pid = fork();
            if (pid == -1)
            {
                printf("\n\nCould not create pipe.\n\n");
            }
            else if (pid > 0)
            {
                {
                    wait(&status);
                    close(pipefd[1]);
                    printf("\nEntered in Parent process\n\n");
                    dup2(pipefd[0], 0);
                    execl("/bin/sort", "sort", (char*)NULL);
                }
            }
            else{
                close(pipefd[0]);
                printf("Entered in Child process\n");
                printf("Enter File name to sort : \n");
                bzero(filenm, sizeof(filenm));
                rd = read(0, filenm, sizeof(filenm));
                filenm[strlen(filenm) - 1] = '\0';
                filehand = open(filenm, O_RDONLY);
                bzero(filedata, sizeof(filedata));
                rd = read(filehand, filedata, sizeof(filedata));
                write(pipefd[1], filedata, rd);
            }
        }
    }
}

```

Output :-



```
dhruvkakadiya@kali: ~/Desktop
File Actions Edit View Help
dhruvkakadiya@kali:~/Desktop$ cat DemoFile.txt
Hello and Welcome to the world of AI!
Recently Elon Musk's company Neuralink represent brain chip!
We have to convert our civilization into type-1 civilization from type 0.7.
that's it!
dhruvkakadiya@kali:~/Desktop$ gcc prog2.c
dhruvkakadiya@kali:~/Desktop$ ./a.out
Entered in Child process
Enter File name to sort :
DemoFile.txt

Entered in Parent process

Hello and Welcome to the world of AI!
Recently Elon Musk's company Neuralink represent brain chip!
that's it!
We have to convert our civilization into type-1 civilization from type 0.7.
dhruvkakadiya@kali:~/Desktop$
```