OS Lab

Name : Gurram Shreya

SRN: PES2UG22CS209

Section: D

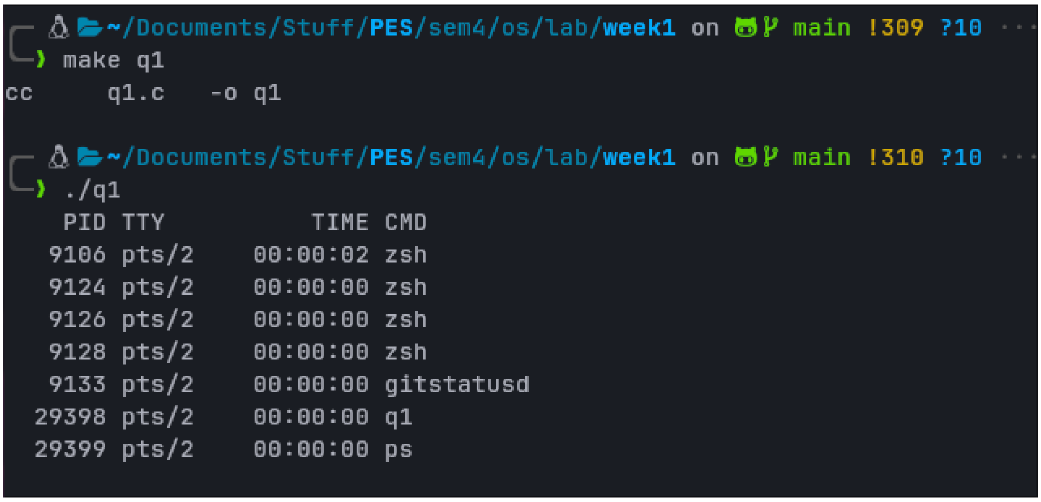
Question 1: Write a program to create a child process which lists all the executing user processes

C

```c
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>

int main() {
  pid_t child = fork();
  if (child == -1)
    return 1;
  if (child) {
    int status;
    waitpid(child, &status, 0);
    return 0;
  } else {
    execl("/usr/bin/ps", "/usr/bin/ps", (char *)NULL);
  }
}
```

Output:



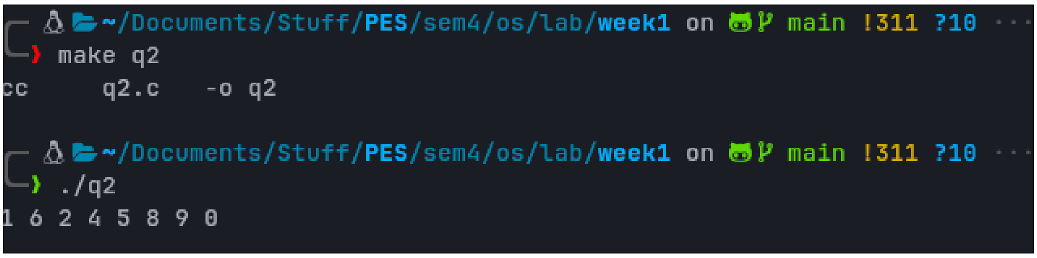
```
╭─ ~/Documents/Stuff/PES/sem4/os/lab/week1 on  main !309 ?10 ···
╰─❯ make q1
cc     q1.c    -o q1

╭─ ~/Documents/Stuff/PES/sem4/os/lab/week1 on  main !310 ?10 ···
╰─❯ ./q1
    PID TTY          TIME CMD
   9106 pts/2    00:00:02 zsh
   9124 pts/2    00:00:00 zsh
   9126 pts/2    00:00:00 zsh
   9128 pts/2    00:00:00 zsh
   9133 pts/2    00:00:00 gitstatusd
  29398 pts/2    00:00:00 q1
  29399 pts/2    00:00:00 ps
```

Question 2: Create a global array with values [1,6,2,4,5,8,9,0]. Sort the same within the child process and display the values in the parent process. Are the displayed values in sorted order? If not,why

```c
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>
int arr[] = {1, 6, 2, 4, 5, 8, 9, 0};
void sort_array() {
  int n = sizeof(arr) / sizeof(arr[0]);
  int temp;
  for (int i = 0; i < n - 1; i++) {
    for (int j = 0; j < n - i - 1; j++) {
      if (arr[j] > arr[j + 1]) {
        temp = arr[j];
        arr[j] = arr[j + 1];
        arr[j + 1] = temp;
      }
    }
  }
}
void display_array() {
  int n = sizeof(arr) / sizeof(arr[0]);
  for (int i = 0; i < n; i++) {
    printf("%d ", arr[i]);
  }
  printf("\n");
}
int main() {
  pid_t pid = fork();
  if (pid == -1) {
    fprintf(stderr, "Failed to fork.\n");
    return 1;
  }
  if (pid == 0) {
    sort_array();
    _exit(0);
  } else {
    wait(NULL);
    display_array();
  }
  return 0;
}
```

Output:



```
╭─ ~/Documents/Stuff/PES/sem4/os/lab/week1 on  main !311 ?10 ···
╰─❯ make q2
cc     q2.c    -o q2

╭─ ~/Documents/Stuff/PES/sem4/os/lab/week1 on  main !311 ?10 ···
╰─❯ ./q2
1 6 2 4 5 8 9 0
```

Answer: The elements in the array are sorted by the child process which can not be accessed by the parent process.

Question 3: Write a program which accepts 2 integers x and y. Now use exec to execute another user defined program that prints the product of x and y.

C

```c
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>

int main() {
  int x, y;
  printf("Enter x and y: ");
  scanf("%d %d", &x, &y);
  pid_t child = fork();
  if (child == -1)
    return 1;
  if (child) {
    int status;
    waitpid(child, &status, 0);
    return 0;
  } else {
    char x_str[10], y_str[10];
    sprintf(x_str, "%d", x);
    sprintf(y_str, "%d", y);
    execl("./q3_helper", "./q3_helper", x_str, y_str, (char *)NULL);
  }
}
```

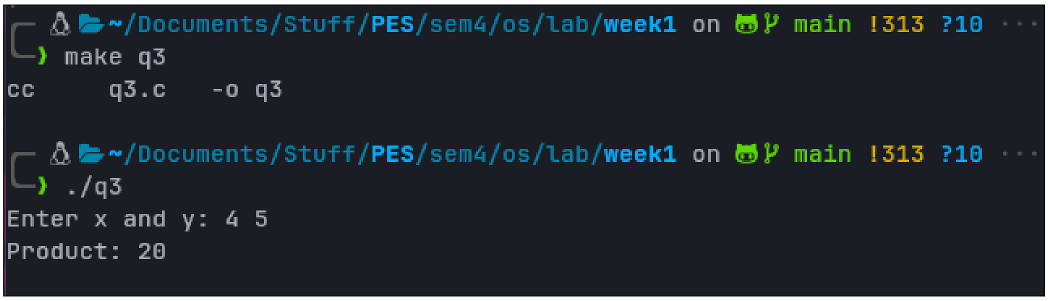
Helper Function:

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

int main(int argc, char **argv) {
  int x = atoi(argv[1]);
  int y = atoi(argv[2]);
  printf("Product: %d\n", x * y);

  return 0;
}
```

Output:



```
  ~/Documents/Stuff/PES/sem4/os/lab/week1 on  main !313 ?10 ···
❯ make q3
cc     q3.c   -o q3

  ~/Documents/Stuff/PES/sem4/os/lab/week1 on  main !313 ?10 ···
❯ ./q3
Enter x and y: 4 5
Product: 20
```