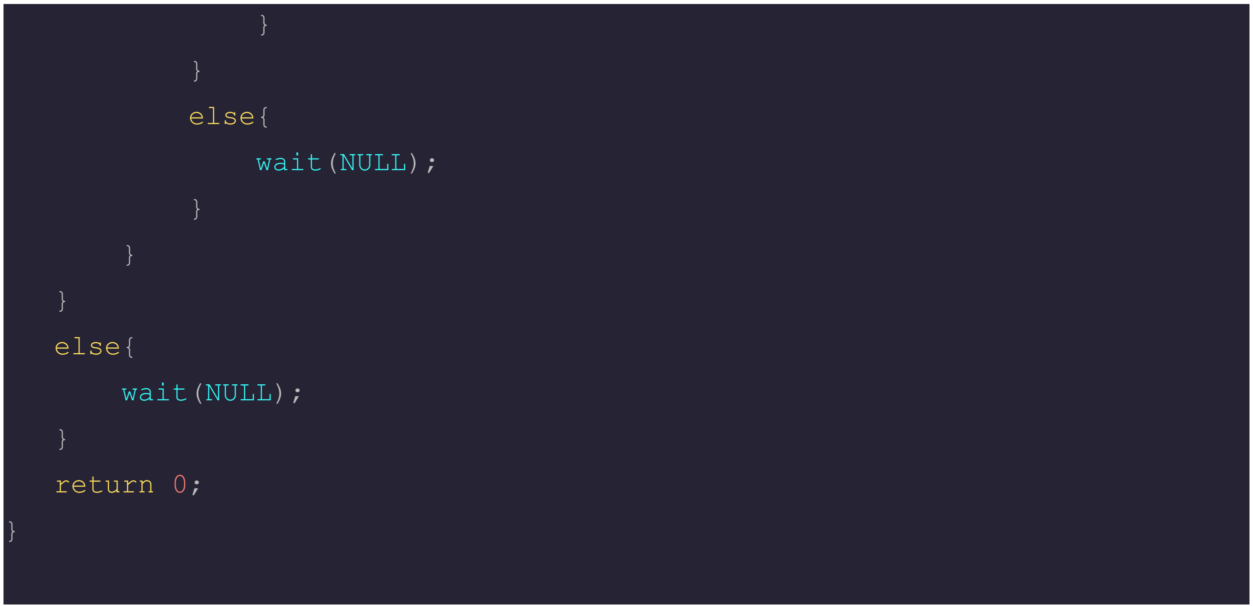
Lab Exercise

Ron Jacob Varghese
AM.EN.U4CSE19346
CSE-D

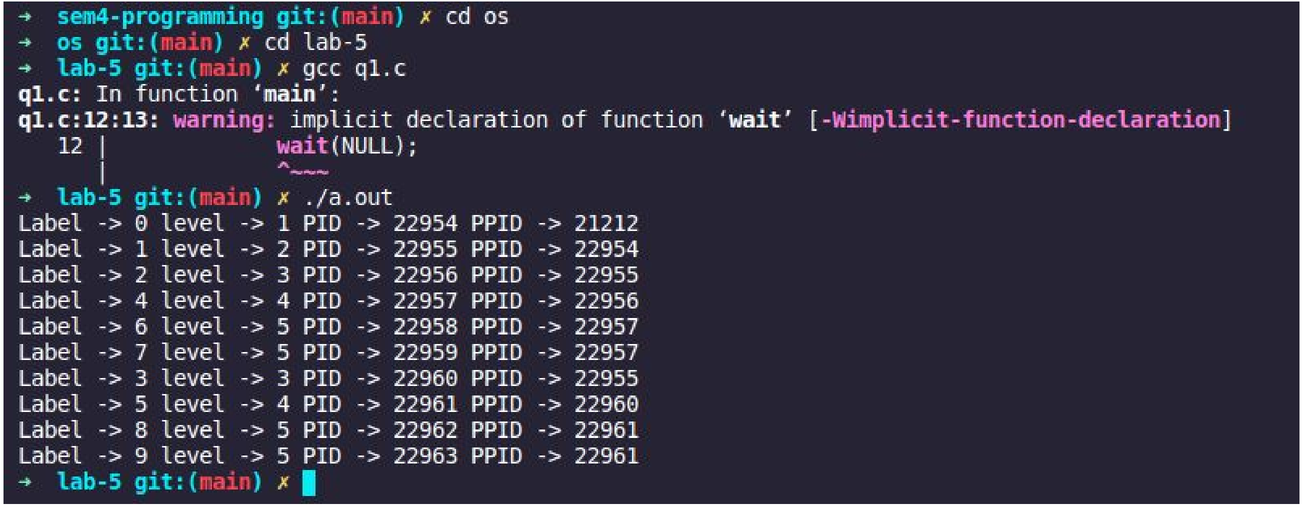
1. The order of the output with the given output is the same(ie A,B,C,E,F,D,G,H,I).When the parent process runs process initially runs, it first calls the wait function so the child process is executed. Hence the order of the execution of the parent and the child will not affect the order in which they are printed.

```
2.#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>

int main()
{
   printf("Label -> 0 level -> 1 PID -> %d PPID -> %d\n", getpid(),
getppid());
   if(!fork())
   {
       printf("Label -> 1 level -> 2 PID -> %d PPID -> %d\n", getpid(),
getppid());
       if(fork()){
           wait(NULL);
           if(!fork()){
               printf("Label -> 3 level -> 3 PID -> %d PPID ->
%d\n",getpid(),getppid());
               if(!fork()) {
                   printf("Label -> 5 level -> 4 PID -> %d PPID -> %d\n",
getpid(), getppid());
                   if(fork()){
                       wait(NULL);
                       if(!fork()){
                           printf("Label -> 9 level -> 5 PID -> %d PPID ->
%d\n",getpid(),getppid());
```

```
                        }
                        else{
                            wait(NULL);
                        }
                    }
                    else{
                        printf("Label -> 8 level -> 5 PID -> %d PPID -> 
%d\n",getpid(),getppid());
                    }
                }
                else
                    wait(NULL);
            }
            else{
                wait(NULL);
            }
        }
        else{
            printf("Label -> 2 level -> 3 PID -> %d PPID -> %d\n", 
getpid(), getppid());
            if(!fork()){
                printf("Label -> 4 level -> 4 PID -> %d PPID -> %d\n", 
getpid(), getppid());
                if(fork()){
                    wait(NULL);
                    if(!fork()){
                        printf("Label -> 7 level -> 5 PID -> %d PPID -> 
%d\n", getpid(), getppid());
                    }
                    else{
                        wait(NULL);
                    }
                }
                else{
                    printf("Label -> 6 level -> 5 PID -> %d PPID -> %d\n", 
getpid(), getppid());
```

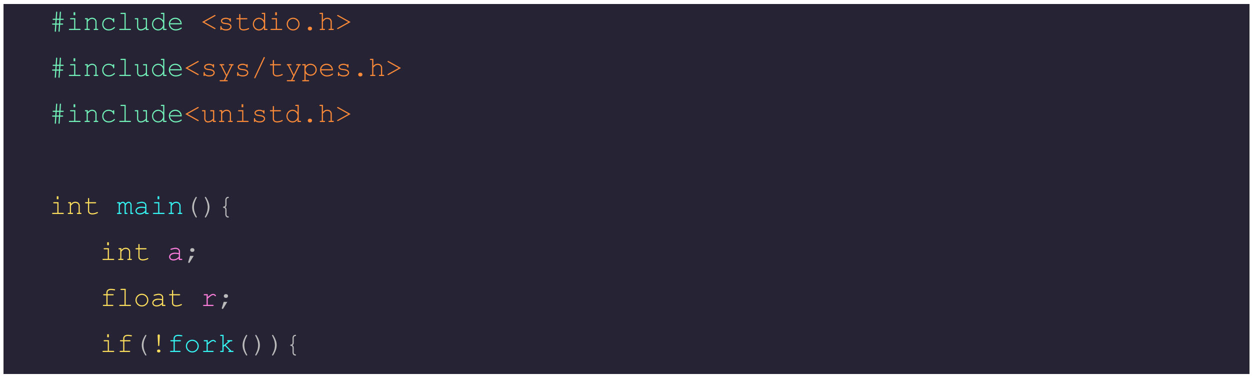


```
                }
            }
            else{
                wait(NULL);
            }
        }
    }
    else{
        wait(NULL);
    }
    return 0;
}
```

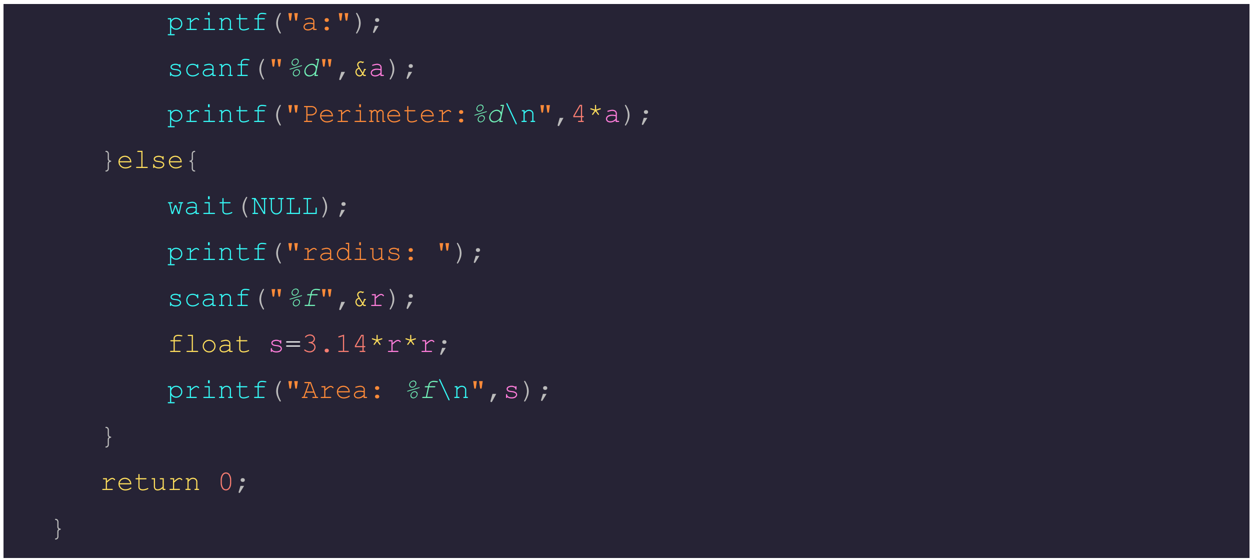


```
➜  sem4-programming git:(main) ✗ cd os
➜  os git:(main) ✗ cd lab-5
➜  lab-5 git:(main) ✗ gcc q1.c
q1.c: In function ‘main’:
q1.c:12:13: warning: implicit declaration of function ‘wait’ [-Wimplicit-function-declaration]
   12 |             wait(NULL);
      |             ^~~~
➜  lab-5 git:(main) ✗ ./a.out
Label -> 0 level -> 1 PID -> 22954 PPID -> 21212
Label -> 1 level -> 2 PID -> 22955 PPID -> 22954
Label -> 2 level -> 3 PID -> 22956 PPID -> 22955
Label -> 4 level -> 4 PID -> 22957 PPID -> 22956
Label -> 6 level -> 5 PID -> 22958 PPID -> 22957
Label -> 7 level -> 5 PID -> 22959 PPID -> 22957
Label -> 3 level -> 3 PID -> 22960 PPID -> 22955
Label -> 5 level -> 4 PID -> 22961 PPID -> 22960
Label -> 8 level -> 5 PID -> 22962 PPID -> 22961
Label -> 9 level -> 5 PID -> 22963 PPID -> 22961
➜  lab-5 git:(main) ✗
```

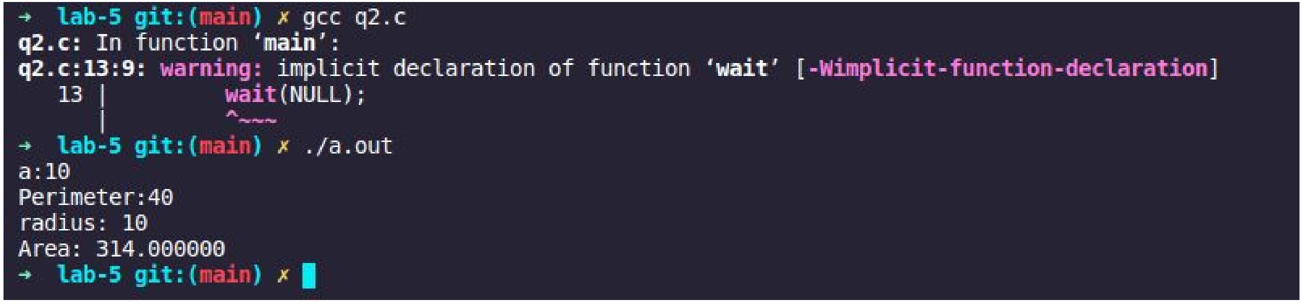
3.



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

int main(){
    int a;
    float r;
    if(!fork()){
```

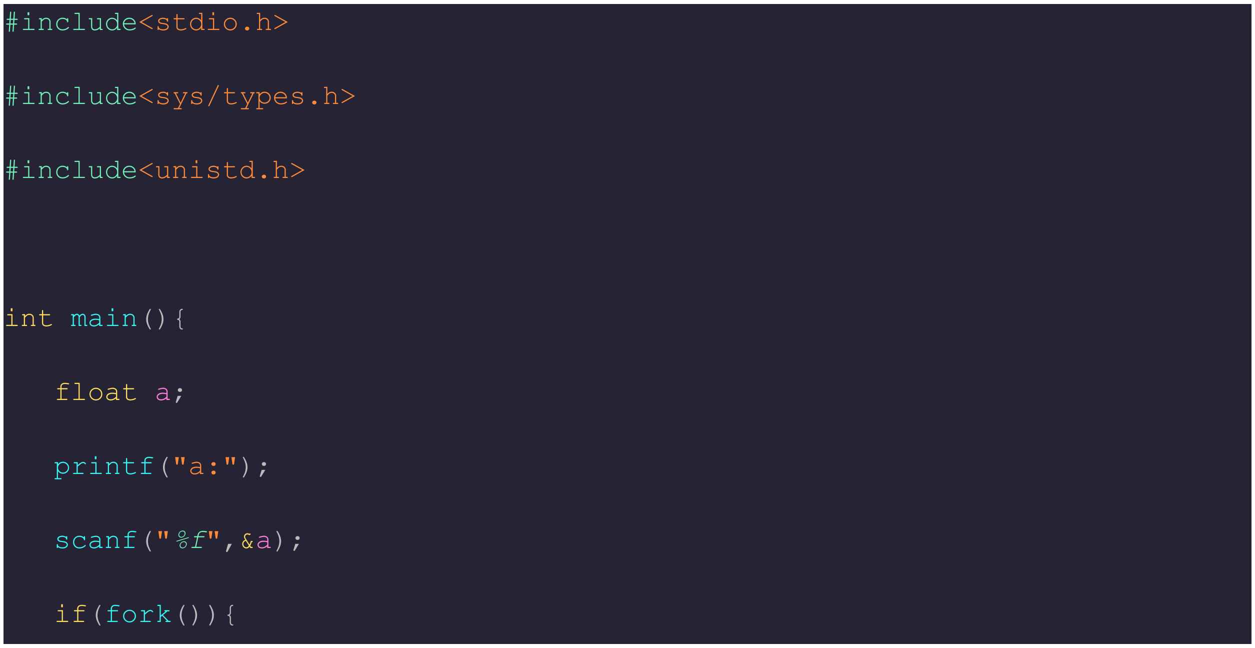


```
        printf("a:");
        scanf("%d",&a);
        printf("Perimeter:%d\n",4*a);
    }else{
        wait(NULL);
        printf("radius: ");
        scanf("%f",&r);
        float s=3.14*r*r;
        printf("Area: %f\n",s);
    }
    return 0;
}
```

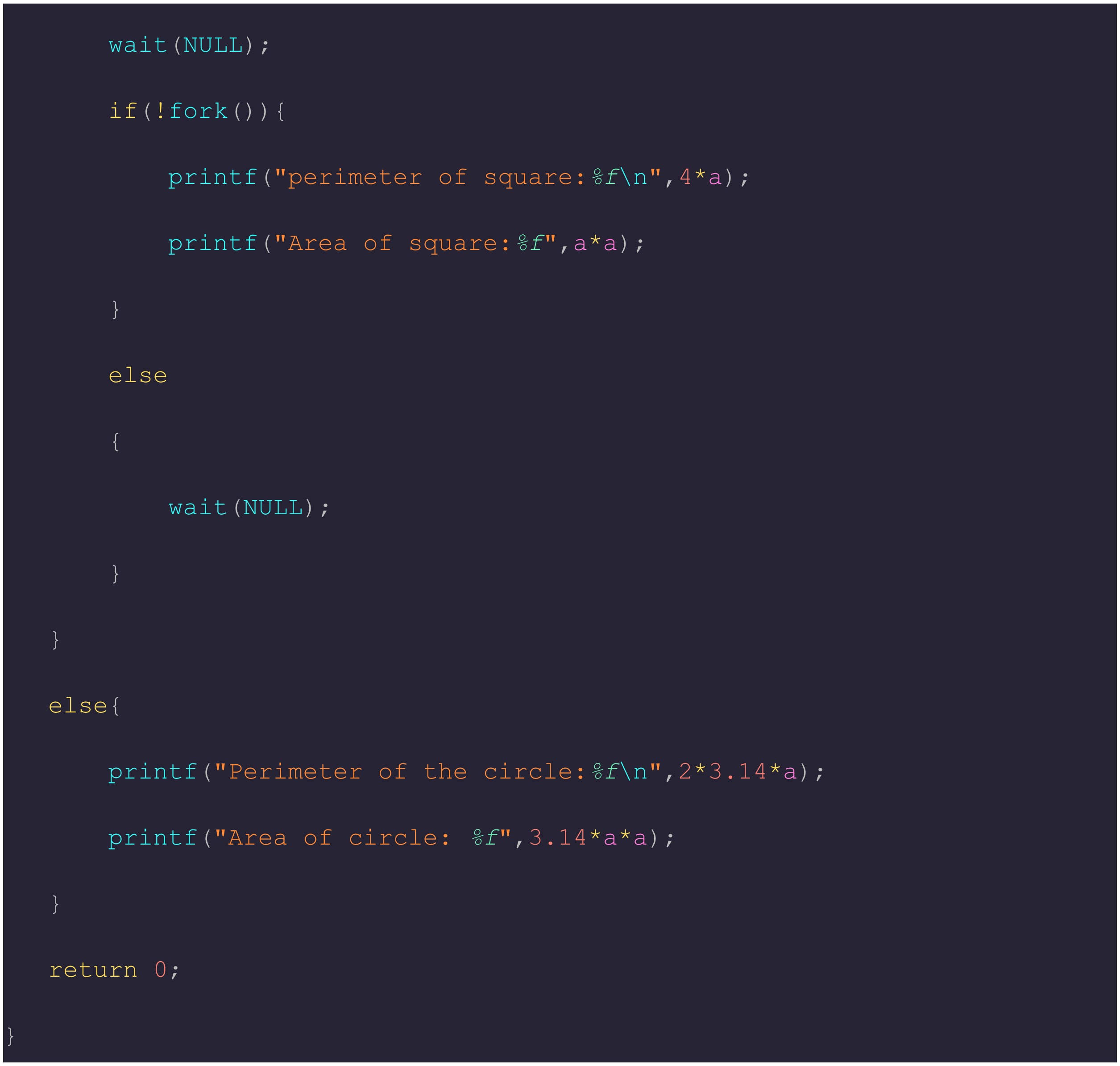


```
➜  lab-5 git:(main) ✗ gcc q2.c
q2.c: In function ‘main’:
q2.c:13:9: warning: implicit declaration of function ‘wait’ [-Wimplicit-function-declaration]
   13 |         wait(NULL);
      |         ^~~~
➜  lab-5 git:(main) ✗ ./a.out
a:10
Perimeter:40
radius: 10
Area: 314.000000
➜  lab-5 git:(main) ✗ 
```

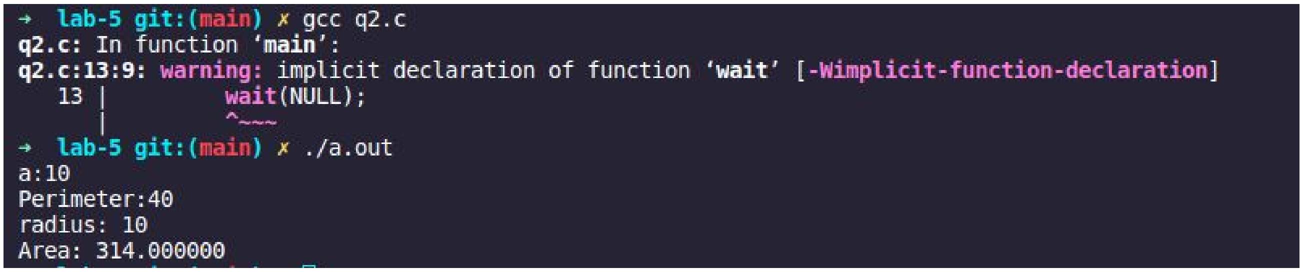
4.



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

int main(){
    float a;
    printf("a:");
    scanf("%f",&a);
    if(fork()){
```

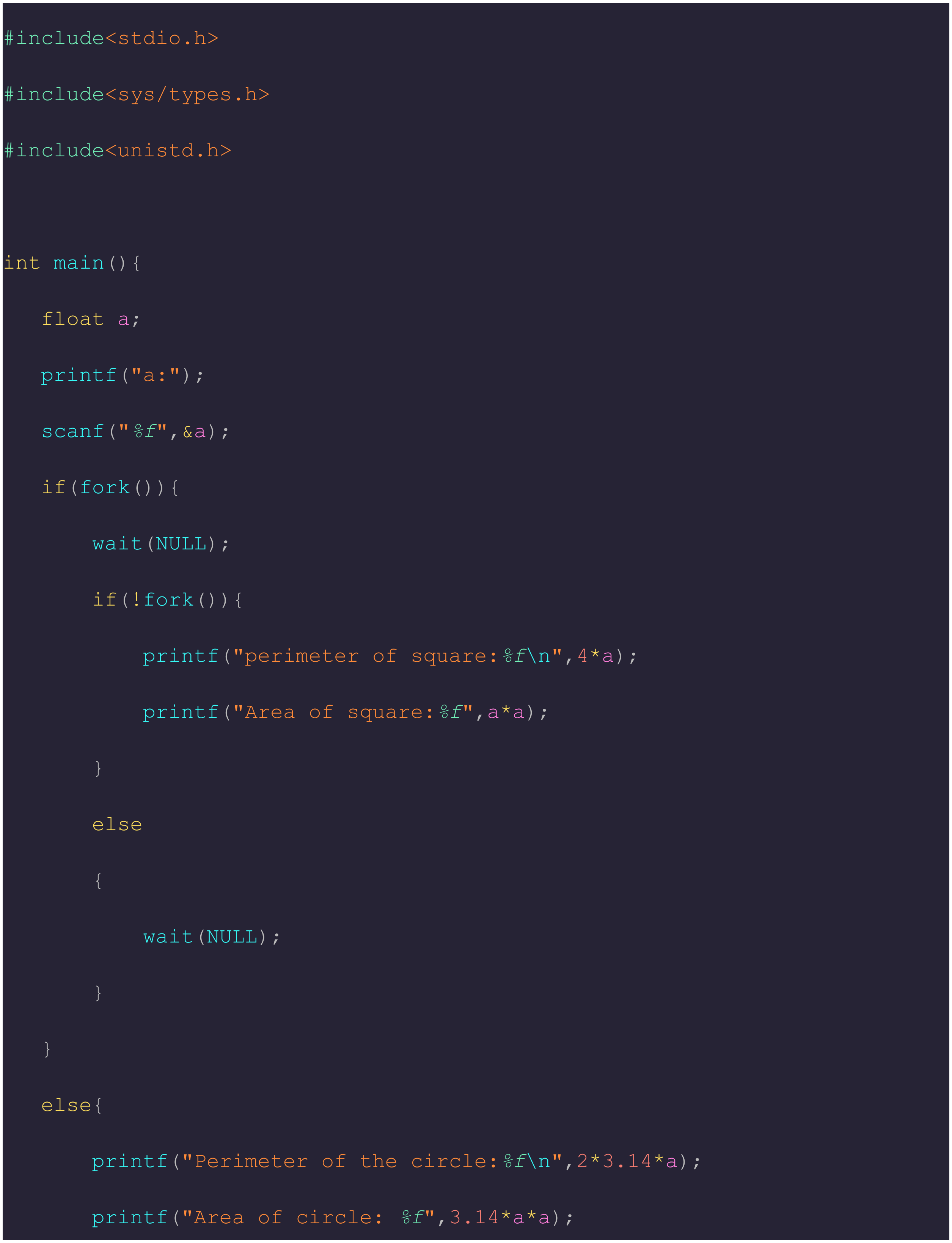


```
        wait(NULL);

        if(!fork()){

            printf("perimeter of square:%f\n",4*a);

            printf("Area of square:%f",a*a);

        }

        else

        {

            wait(NULL);

        }

    }

    else{

        printf("Perimeter of the circle:%f\n",2*3.14*a);

        printf("Area of circle: %f",3.14*a*a);

    }

    return 0;

}
```

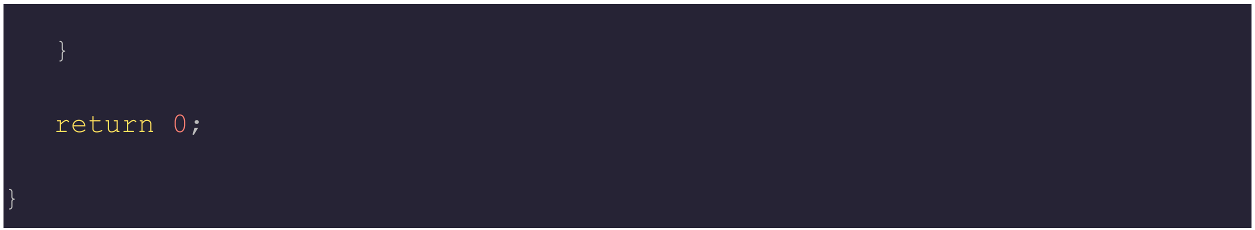


```
→  lab-5 git:(main) ✗ gcc q2.c
q2.c: In function ‘main’:
q2.c:13:9: warning: implicit declaration of function ‘wait’ [-Wimplicit-function-declaration]
   13 |         wait(NULL);
      |         ^~~~
→  lab-5 git:(main) ✗ ./a.out
a:10
Perimeter:40
radius: 10
Area: 314.000000
```

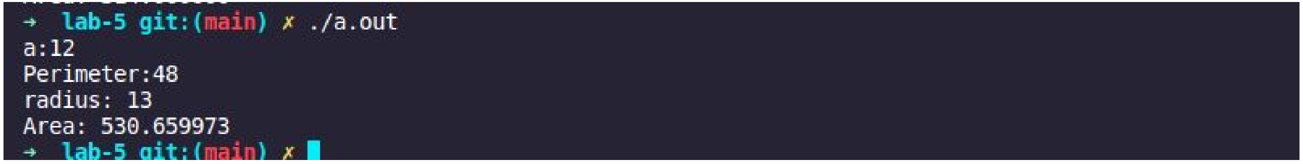
5.



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

int main(){
   float a;
   printf("a:");
   scanf("%f",&a);
   if(fork()){
       wait(NULL);
       if(!fork()){
           printf("perimeter of square:%f\n",4*a);
           printf("Area of square:%f",a*a);
       }
       else
       {
           wait(NULL);
       }
   }
   else{
       printf("Perimeter of the circle:%f\n",2*3.14*a);
       printf("Area of circle: %f",3.14*a*a);
```

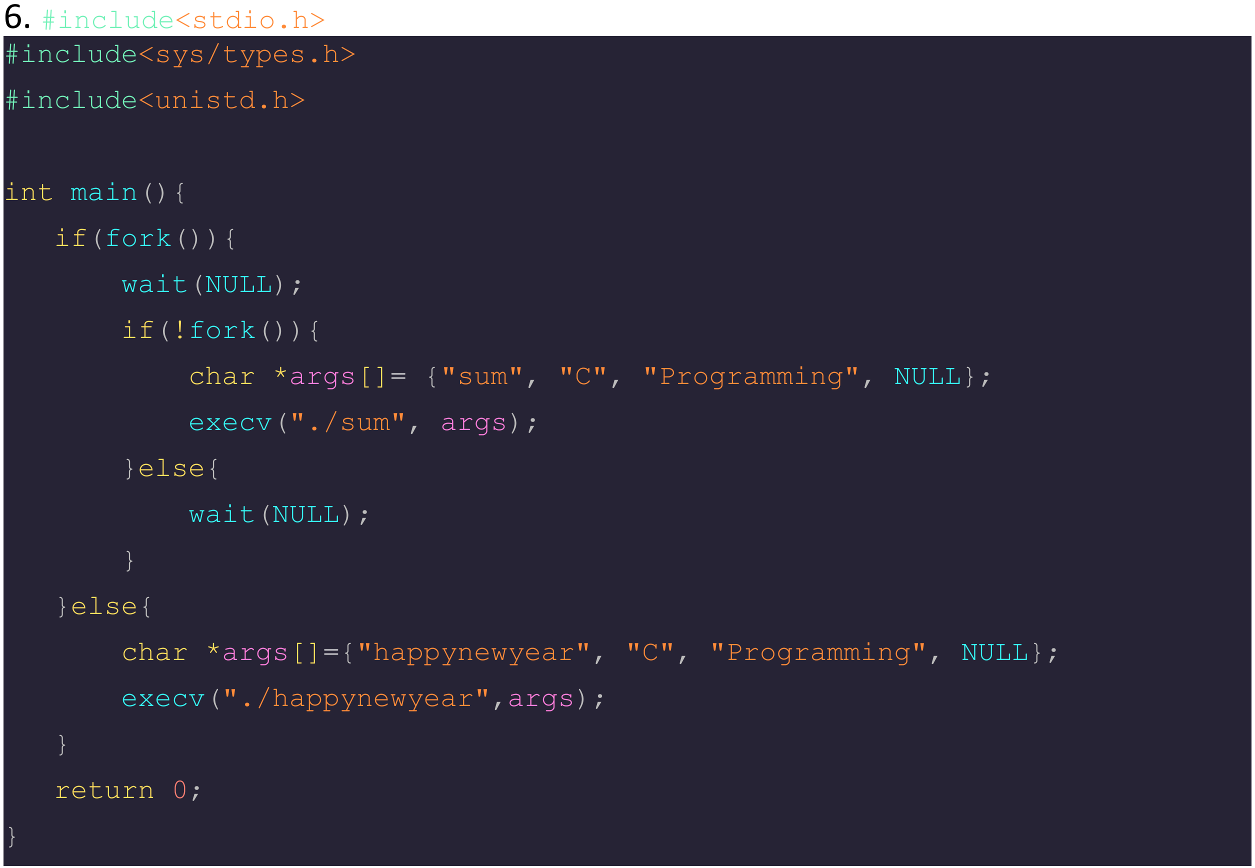


```
    }

    return 0;

}
```

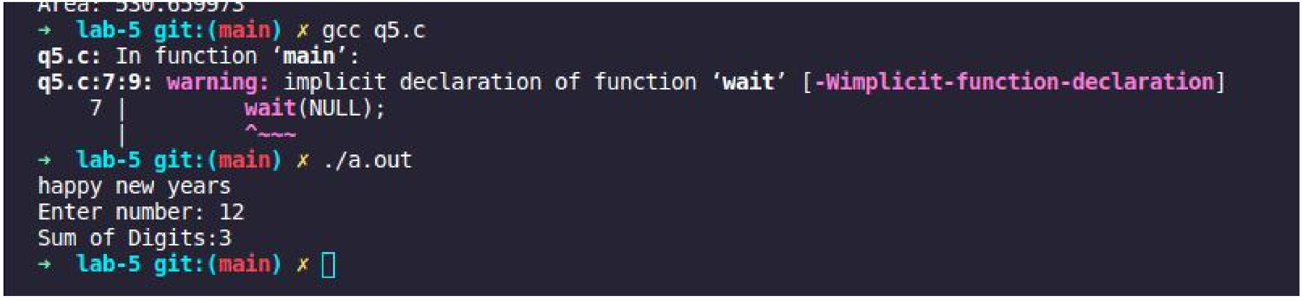


```
 ➜  lab-5 git:(main) ✗ ./a.out
 a:12
 Perimeter:48
 radius: 13
 Area: 530.659973
 ➜  lab-5 git:(main) ✗ █
```

6.



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

int main(){
    if(fork()){
        wait(NULL);
        if(!fork()){
            char *args[]= {"sum", "C", "Programming", NULL};
            execv("./sum", args);
        }else{
            wait(NULL);
        }
    }else{
        char *args[]={"happynewyear", "C", "Programming", NULL};
        execv("./happynewyear",args);
    }
    return 0;
}
```



```
 Area: 530.659973
 ➜  lab-5 git:(main) ✗ gcc q5.c
 q5.c: In function ‘main’:
 q5.c:7:9: warning: implicit declaration of function ‘wait’ [-Wimplicit-function-declaration]
     7 |         wait(NULL);
       |         ^~~~
 ➜  lab-5 git:(main) ✗ ./a.out
 happy new years
 Enter number: 12
 Sum of Digits:3
 ➜  lab-5 git:(main) ✗ ▯
```