Fundamentals of Computing and Programming Final Examination

Date: 18^{th} November 2024 Time: 3 hours Total marks: 50

PART A (15 marks)

This part has 5 questions. Please answer all questions. Each question carries 3 marks. Please read the programs carefully before answering them.

Question 1

What would be the output of the following program.

```
#include <stdio.h>
int compute(int x, int y)
{
    int z;
    while ((z = (x \% y)) != 0)
    {
        x = y;
        y = z;
    }
    return y;
}
int main()
{
    printf("%d\n", compute(12, 32));
    printf("%d\n", compute(13, 32));
    printf("%d\n", compute(91, 133));
}
```

What is the output of the following program?

```
#include <stdio.h>
void f(int n)
{
   n += 1;
}
void g(int *n)
{
   *n += 1;
}
int main()
{
   int x = 10;
   int y = 20;
    f(x);
   g(&y);
   printf("%d %d\n", x, y);
}
```

What is the output of the following program?

```
#include <stdio.h>
void move(int n, int x, int y, int z)
{
    if (n==1)
    {
        printf("%d -> %d\n", x, y);
    }
    else
    {
        move(n-1, x, z, y);
        printf("%d -> %d\n", x, y);
        move(n-1, z, y, x);
     }
}
int main(void)
{
    move(3, 1, 2, 3);
    return 0;
}
```

What is the output of the following program?

```
#include <stdio.h>
int main()
{
    int a = 0;
    int b = 1;
    int tmp;
    while (b < 100)
    {
        tmp = b;
        b = a + b;
        a = tmp;
        printf("%d\n", a);
    }
}</pre>
```

What is the output of the following program?

```
int f(int n)
{
    int i;
    int result = 0;
    for (i=1; i<=n; i++)
    {
        result += i;
        return result;
    }
}
int main()
{
    int n = f(10);
    printf("%d\n", n);
}</pre>
```

PART B (15 marks)

This part as 3 questions. Please answer all of them. Each question carries 5 marks.

Question 6

Write a program countdown.c that takes a positive integer as a command-line argument and prints all numbers counting down from that number to 1.

For example:

```
$ gcc countdown.c -o countdown
$ ./countdown 5
5
4
3
2
1
```

Problem 7

Write a function contains to test if a point falls inside a circle. The function takes a circle and a point as arguments and returns 1 if the point falls inside the circle (includes the boundary), returns 0 if the point is outside the circle.

```
struct point
{
    float x
    float y
};
struct circle
{
    struct point center
    float radius
};
// your code will come here
int main()
{
    struct circle c = \{\{0, 0\}, 5\};
    struct point p1 = {1, 1};
    struct point p2 = \{3, 4\};
    struct point p3 = \{4, 4\};
    printf("%d\n", contains(p1, c)); // prints 1
    printf("%d\n", contains(p2, c)); // prints 0
    printf("%d\n", contains(p3, c)); // prints 0
}
```

Please write the answer in the next page.

Problem 8

Write a function **array_reverse** that takes an array of integer and size of the array as arguments and reverse the order of elements. The function should be able to take array of any size.

For example, consider the following program:

```
#include <stdio.h>
// your code will come here
int main()
{
    int a[4] = \{1, 2, 3, 4\};
    int b[7] = {1, 2, 3, 4, 5, 6, 7};
    int i;
    // reverse a and print it
    array_reverse(a, 4);
    for (i=0; i<4; i++)</pre>
    {
        printf("%d ", a[i]);
    }
    printf("\n");
    // reverse b and print it
    array_reverse(b, 7);
    for (i=0; i<7; i++)</pre>
    {
        printf("%d ", b[i]);
    }
    printf("\n");
}
```

After including your implementation of **array_reverse**, the above program would output the following:

4 3 2 1 7 6 5 4 3 2 1

Please write the answer in the next page.

PART C (20 marks)

This part has three questions. Please answer any **two** of them. Each question carries 10 marks.

Question 9

Let n be a positive integer. Let d(n) be defined as the sum of proper positive divisors of n, the positive numbers less than n which divide n.

The pair of positive numbers a and b where $a \neq b$ are called an *amicable pair* if d(a) = b and d(b) = a, and each of the numbers a and b are called *amicable numbers*.

For example, the pair of numbers 220 and 284 is an amicable pair since:

d(220) = 1 + 2 + 4 + 5 + 10 + 11 + 20 + 22 + 44 + 55 + 110 = 284, and d(284) = 1 + 2 + 4 + 71 + 142 = 220

Write a program to compute the sum of all amicable numbers between 1 and 10,000.

Twin primes are defined to be two consecutive odd numbers which are both primes. For example, 11 and 13 are twin primes, and so are 17 and 19. Write a program to generate all twin primes from 1 to 1,00,000.

The following iterative sequence is defined for the set of positive integers:

 $n \to n/2$, if n is even

 $n \rightarrow 3n + 1$, if n is odd

Using the above rules and starting with 13, we generate the following sequence of numbers, terminating when the number 1 is reached.

 $13 \rightarrow 40 \rightarrow 20 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1.$

It can be seen that this sequence which starts at 13 and ends at 1 contains 10 terms. It is conjectured (Collatz Problem) that all starting numbers must reach 1.

Write a program to find the smallest postive integer under 1,000, which produces the longest such sequence of numbers?

Please note that once the sequence starts the terms are allowed to go above one thousand, only the starting number must be below one thousand.