CS 1713
Introduction to Computer Programming II
Midterm

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question 1</td>
<td></td>
</tr>
<tr>
<td>Question 2</td>
<td></td>
</tr>
<tr>
<td>Question 3</td>
<td></td>
</tr>
<tr>
<td>Question 4</td>
<td></td>
</tr>
<tr>
<td>Question 5</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
</tr>
</tbody>
</table>

NAME:_________________________

Instructions
1. Do all of the 5 problems
3. You have 70 minutes for the exam
4. Show all your work
5. Do not separate midterm papers

Easy               Difficulty Level               Difficult
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐
1 2 3 4 5 6 7 8 9 10
1. (20 pts) Complete the following program to find the average and standard deviation of the numbers in an array. For example, the array \( \{2, 4, 6, 3, 9, 10\} \) has standard deviation of average of 5.666667 and standard deviation of 2.981424. Formula for standard deviation is as follows.

\[
\text{stddev} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - x_{avg})^2}
\]

In the formula \( n \) denotes the number of items in the set, \( x_i \) denotes \( i^{th} \) number and \( x_{avg} \) denotes the average of the numbers in the set.

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

int main()
{
    double stddev;  //standard deviation
    int i;
    double num[6]; // array to store the numbers
    double average, double sum=0;

    printf("Enter 6 doubles\n");
    for (i=0; i<6; i++)
        scanf("%lf", &num[i]);
    for (i = 0 ; i < 6 ; i++)
        sum = sum + num[i];
    average = sum / 6 ;
    Sum=0
    for (i=0; i<6; i++)
        sum = sum + ((num[i]-average) * (num[i]-average));
    stddev = sqrt((sum / 6));

    printf("Average is %lf\n",average);
    printf("Standard Deviation is %lf\n",stddev);
    return(0);
}
```
2. (20 pts) Trace the execution of the following program. What will be the final values of array \( a \) printed?

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

int main()
{
    int a[5]={1,2,3,4,5};
    int i,j,temp;

    for (j=1; j<5; j++)
        for (i=0; i<5-j; i=i+2)
        {
            printf("%d %d\n",i,i+j);
            temp = a[i];
            a[i] = a[i+j];
            a[i+j] = temp;
        }

    for (i=0; i<5; i++)
        printf("a[%d] = %d\n",i,a[i]);
}
```


Trace:

- \( i=0, j=1 \):
  - \( a[0] \) and \( a[2] \) swapped:
    - \( a[0]=2 \), \( a[2]=1 \)

- \( i=1, j=2 \):
  - \( a[1] \) and \( a[3] \) swapped:
    - \( a[1]=4 \), \( a[3]=2 \)

- \( i=2, j=3 \):
  - \( a[2] \) and \( a[4] \) swapped:
    - \( a[2]=5 \), \( a[4]=3 \)

Final values: \( a[0]=2 \), \( a[1]=4 \), \( a[2]=5 \), \( a[3]=1 \), \( a[4]=3 \)
3. (20 pts) What is the output of the following program? Show all your work for partial credit.

```c
#include <stdio.h>

int function1(int a, int b)
{
    return(a+2*b);
}

int function2(int a)
{
    return(2*a+1);
}

int main()
{
    int i=2;
    int x;

    while (i<10)
    {
        if (i>5)
        {
            x = function1(i+1,i+1);
        }
        else
        {
            x = function2(i+1)-i;
        }
        printf("%d\n",x);
        i = i + 1;
    }

    return(0);
}
```

\[
\begin{align*}
X &= \text{function2}(3) - 2 \\
   &= 2 \times 3 + 1 - 2 = 5 \\
X &= \text{function2}(4) - 3 \\
   &= 2 \times 4 + 1 - 3 = 6 \\
X &= \text{function2}(5) - 4 \\
   &= 2 \times 5 + 1 - 4 = 7 \\
X &= \text{function2}(6) - 5 \\
   &= 2 \times 6 + 1 - 5 = 8 \\
X &= \text{function2}(7) - 7 \\
   &= 7 + 2 \times 7 = 21 \\
X &= \text{function2}(8) - 8 \\
   &= 8 + 2 \times 8 = 24 \\
X &= \text{function2}(9) - 9 \\
   &= 9 + 2 \times 9 = 27 \\
\end{align*}
\]
4. (30 pts) Write a complete program to compute the following expression. Read the value of \( n \) from the user and write a single loop to evaluate the expression. Do not use \texttt{pow} function in your program.

\[
\sum_{i=1}^{n} \frac{1}{2^i} - \sum_{i=1}^{n} \frac{1}{3^i}
\]
CS 1713
Introduction to Computer Programming II
Midterm

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question 1</td>
<td></td>
</tr>
<tr>
<td>Question 2</td>
<td></td>
</tr>
<tr>
<td>Question 3</td>
<td></td>
</tr>
<tr>
<td>Question 4</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
</tr>
</tbody>
</table>

NAME:

Instructions
1. Do all of the 4 problems
3. You have 50 minutes for the exam
4. Show all your work
5. Do not separate midterm papers

<table>
<thead>
<tr>
<th>Easy</th>
<th>Difficulty Level</th>
<th>Difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td>☐</td>
<td>☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐</td>
<td>☐ ☐ ☐ ☐</td>
</tr>
</tbody>
</table>
1. (30 pts) Complete the following program to find if there are two numbers in the array whose sum is 10. For example, the array \{1, 2, 3, 4, 5, 6\} has 4 and 6 at index positions 3 and 5 with respectively with sum 10. If there are multiple pairs with sum 10, just print one of them.

```
#include <stdio.h>
#include <math.h>

int main()
{
    int i, j;
    double num[6];
    double sum = 10;
    int found = 0, first, second;

    printf("Enter 6 doubles\n");
    for (i=0; i<6; i++)
        scanf("%lf", &num[i]);
    for (i = 0; i < 6; i++)
        for (j = i+1; j < 6; j++)
            if (num[i] + num[j] == sum)
                found = 1,
                first = i;
                second = j;

    if (found == 1)
        printf("num[%d] and num[%d] has desired sum", first, second);
    else
        printf("none of the pairs have desired sum\n");
    return(0);
}
```
2. (20 pts) Trace the execution of the following program? What will be the final values of array a printed for the cases given below.

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

int main()
{
    int a[7]; // see text below;
    int i=2;
    while (i<7)
    {
        if (i<5)
            a[i] = a[i-1]+a[i-2];
        else
            a[i] = a[i-3];
        i = i + 1;
        for (i=0; i<7; i++)
            printf("a[%d] = %d
", i, a[i]);
    }
    return 0;
}
```

(a) What will be the final array when initial array is $a[7] = \{1, 1, 0, 0, 0, 0, 0\}$?

- $a[2] = a[1] + a[0]$

(a) $a[7] = 5$

(b) What will be the final array when initial array is $a[7] = \{4, 2, 3, 6, 4, 2, 5\}$?


(b) $a[7] = 10$
3. (20 pts) Write a function `Powerof3` to test if a parameter `n` is a power of 3 (`n = 3^k` for some integer `k`). If `n` is a power of 3, then the function returns 1. Otherwise it returns 0. Function prototype and sample output of the function and description of the output is given below.

   for n=5 function returns 0 since $3^1 = 3 < 5 < 3^2 = 9$
   for n=9 function returns 1 since $9 = 3^2$
   for n=30 function returns 0 since $3^3 = 27 < 30 < 3^4 = 81$
   for n=27 function returns 1 since $27 = 3^3$

```c
int Powerof3(int n)
{
}
```
4. (20 pts) What is the output of the following program? Show all your work for partial credit.

```c
#include <stdio.h>

int function1(int x)
{
    return(2*x+1);
}

int main()
{
    int j=3;

    while (j < 50)
    {
        if (j<10)
            j = function1(j-1);
        else
            j = function1(j)-1;
        printf("%d\n",j);
    }

    return(0);
}
```
5. (20 pts) Write a complete program to compute the following expression. Read the value of $n$ from the user and write a loop to evaluate the product. $\prod$ denotes multiplication of all the terms.

$$\frac{1}{2 \times 1 - 1} \times \frac{2}{2 \times 2 - 1} \times \cdots \times \frac{i}{2 \times i - 1} \times \cdots \times \frac{n}{2 \times n - 1} = \prod_{i=1}^{n} \frac{i}{2i - 1}$$
CS 1713
Introduction to Computer Programming II
Midterm

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question 1</td>
<td></td>
</tr>
<tr>
<td>Question 2</td>
<td></td>
</tr>
<tr>
<td>Question 3</td>
<td></td>
</tr>
<tr>
<td>Question 4</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
</tr>
</tbody>
</table>

NAME: __________________________

Instructions
1. Do all of the 4 problems
3. You have 50 minutes for the exam
4. Show all your work
5. Do not separate midterm papers

<table>
<thead>
<tr>
<th>Easy</th>
<th>Difficulty Level</th>
<th>Difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td>☐</td>
<td>☐ ☐ ☐ ☐ ☐ ☐ ☐</td>
<td>☐ ☐ ☐ ☐</td>
</tr>
<tr>
<td>1</td>
<td>2 3 4 5 6 7 8 9</td>
<td>10</td>
</tr>
</tbody>
</table>
1. (30 pts) Complete the following program to find the difference of a set of numbers in an array. Difference of a set of numbers is the difference between largest element and smallest element in the set. For example, the array \{2, 4, 6, 3, 9, 10\} has difference of 8 since largest element is 10 and smallest element is 2 resulting in difference of 10-2=8.

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

int main()
{
    int i;
    double num[6];
    double max;
    double min;
    double difference;

    printf("Enter 6 doubles\n");
    for (i=0; i<6; i++)
        scanf("%lf", &num[i]);

    printf("Difference is %lf\n", difference);
    return(0);
}
```
2. (20 pts) Trace the execution of the following program. What will be the final values of array $a$ printed?

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

int main()
{
    int a[7]={1,2,4,8,16,32,64};
    int i;

    for (i=1; i<6; i=i+2)
    {
        if (i<4)
        {
            a[i] = (a[i-1]+a[i+1])/2;
            a[i-1] = a[i]+1;
        }
        else
        {
            a[i] = a[6-i];
            a[i-1] = a[i]-1;
        }
    }

    for (i=0; i<7; i++)
        printf("a[%d] = %d\n", i, a[i]);
}
```
3. (20 pts) What is the output of the following program? Show all your work for partial credit.

```c
#include <stdio.h>

int function1(int a, int b)
{
    return((a-1)*(b+1)/2);
}

int main()
{
    int i;
    int x;

    for (i=2; i<10; i=i+1)
    {
        if (i>10)
            x = function1(i, i-1);
        else
            x = function1(i-1, i);
        printf("%d\n", x);
    }

    return(0);
}
```

\[
x = \text{function1}(1, 2) = (1-1)*(2+1)/2 = 0
\]

\[
x = \text{function1}(2, 3) = (2-1)*(3+1)/2 = 2 \times 4/2 = 2
\]

\[
x = \text{function1}(3, 4) = (3-1)*(4+1)/2 = 5
\]
4. (30 pts) Write a **complete program** to compute the following expression. Read the value of \( n \) from the user and write a **single loop** to evaluate the expression.

\[
\sum_{i=1}^{n} \frac{1}{i} + \sum_{i=1}^{n} \frac{1}{2^i} + \frac{1}{n} + \frac{1}{2*1} + \frac{1}{2*2} + \frac{1}{2*3} + \cdots + \frac{1}{2*n}
\]

```c
#include <stdio.h>

int main()
```

```c
    int i, n;
    float sum = 0;
    printf("Enter n \n");
    scanf("%d", &n);
    
    for (i = 1; i <= n; i++)
        sum += 1.0 / i + 1.0 / (2 * i);
    
    printf("Sum = %f\n", sum);
    return 0;
```