CS 1713
Introduction to Computer Programming II
Midterm Solutions

<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
antiago 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.

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

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.

**Solution:**

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

int main()
{
    double stddev; //standard deviation
    int i;
    double num[6];
    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 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\n",i,a[i]);
}
```

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

**Solution:**

\[
\begin{align*}
a[0] & = 1 \\
a[1] & = 1 \\
a[2] & = 2 \\
a[3] & = 3 \\
a[4] & = 5 \\
a[5] & = 2 \\
a[6] & = 3 \\
\end{align*}
\]

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

**Solution:**

\[
\begin{align*}
a[0] & = 4 \\
a[1] & = 2 \\
a[2] & = 6 \\
a[3] & = 8 \\
a[4] & = 14 \\
a[5] & = 6 \\
a[6] & = 8 \\
\end{align*}
\]
3. (20 pts) Write a function \textit{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 \(27 = 3^3\)

**Solution:**

```c
int Powerof3(int n)
{
    int i=1,power;

    power = 3;
    while (power<n)
    {
        i = i + 1;
        power = power * 3;
    }

    if (power == n)
        return(1);
    else
        return(0);
}
```
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);
}

Solution:

5
9
17
34
68
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.

\[
\frac{1}{2*1-1} \cdot \frac{2}{2*2-1} \cdot \ldots \cdot \frac{i}{2*i-1} \cdot \ldots \cdot \frac{n}{2*n-1} = \prod_{i=1}^{n} \frac{i}{2i-1}
\]

Solution:

```c
#include <stdio.h>

int main()
{
    int n,i;
    float product = 1;

    printf("Enter n\n");
    scanf("%d",&n);

    for (i=1; i<=n; i=i+1)
        product = product * i/(2.0*i-1);

    printf("Product = %f\n",product);

    return(0);
}
```