

CS10003 Programming and Data Structures

Spring Semester, 2021-2022

Sections 9 & 10

Long Test 1 – Part A [Marks = 50]

18-May-2022, 9:00 to 10:10

INSTRUCTIONS

- You have 1 hour for writing and 10 minutes for submission.
 - Write your answers on paper. Answers must be handwritten. Typed or written answers using an electronic device are not allowed.
 - Write your name and roll number on each page. Write page numbers for each page.
 - Scan all pages and collate. Create a single PDF file (of size ≤ 10 MB). You could also take pictures of different pages, combine them to make a single pdf file.
 - The name of the files for part A should be `<RollNumber>_Long1A`. Ensure your roll number in the filename has only digits and uppercase characters.
 - Upload your file. Make sure you click on the 'Turn in' button to submit your file.
 - The said deadlines are strict after which no submissions will be allowed.
 - We will not accept submission by any other means.
-

1. Write a C program that reads four floating point numbers x_1, y_1, x_2, y_2 and determines whether the line segment joining the points (x_1, y_1) and (x_2, y_2) intersects the line $3y - x = 2$. If so, prints "Yes" and otherwise prints "No". Your program should have only the `main()` function. Do not use any library functions other than standard input and output. 12

Solution:

```
#include <stdio.h>

int main(){
    float x_1,y_1,x_2,y_2;
    scanf("%f %f %f %f", &x_1,&y_1,&x_2,&y_2);
    int flag = (3*y_1-x_1-2 < 0 && 3*y_2-x_2-2 > 0) || (3*y_1-x_1-2 == 0 && 3*y_2-x_2-2 == 0) ||
    (3*y_1-x_1-2 > 0 && 3*y_2-x_2-2 < 0) || (3*y_1-x_1-2 == 0 && 3*y_2-x_2-2 == 0);
    if(flag)
        printf("Yes\n");
    else
        printf("No\n");
    return 0;
}
```

2. Consider the switch statement below, where a is an int type variable. What will be printed if $a = 1, b = 2$ initially? Briefly, explain the logic of working.

```

switch(a && b) {
    case 1: a = a++;
    case 2: a = ++a; printf("%d", a);
            break;
    case 3: b = a + 50;
            printf("%d", a);
    default: a = a++ + ++a;
            printf("%d", a);
}

```

8

Solution: 2

3. Write a C program that uses iteration in a loop to compute the power tower of a given height n of a given double floating-point number x . The power tower of x of heights 1, 2 and 3 are respectively x , x^x , and x^{x^x} . For example $5^5 = 3125$, $2^{2^2} = 16$, and $2^{2^{2^2}} = 65536$.

10

Solution:

```

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

double g(double x,int n);

int main (){
    double x; int n;
    scanf("%lf %d",&x,&n);
    printf("g(%lf,%d)=%lf\n",x,n,g(x,n));
}

double g(double x,int n)
{
    int i=n; double result=1;
    while (i>0){
        result=pow(x,result);
        i=i-1;
    }
    return(result);
}

```

4. Write a C program to read a positive integer n and find the smallest integer $m > n$ such that n has at least three distinct integer factors other than 1 and n . For example, if $n = 6$, then the answer is 12. Do not use arrays or any library functions other than standard input and output. There should be only one main function and no other functions defined by you.

12

Solution:

```

#include<stdio.h>
main()
{
    int n, m, i, j, done, factcount, numb;
    scanf("%d", &n);
    printf("n = %d \n", n);
    i = n+1;
    done = 0;

```

```

while (done == 0)
{
    factcount = 0;
    for (j = 2; j < i; j++)
    {
        if (i%j == 0) factcount++;
        if (factcount == 3)
        { numb = i;
          done = 1;
          break;
        }
    }
    i++;
}

printf("Smallest number > %d with 3 such factors is %d \n", n,numb);
}

```

5. Write a program that reads in 10 integers from standard input, and prints the fraction of values that are strictly above the average value. Your program should have only the *main()* function and not any other function. 8

Solution:

```

#include <stdio.h>

int main(){
    int a[10];
    int i;
    int avg = 0;
    for(i=0; i<10; i++){
        scanf("%d", &a[i]);
        avg += a[i];
    }
    avg = avg/10;

    int k = 0;
    for(i=0; i<10; i++){
        if(a[i] > avg)
            k++;
    }
    printf("Fraction of values greater than the average: %g\n", k/10.0);
    return 0;
}

```