



INDIAN INSTITUTE OF TECHNOLOGY
KHARAGPUR

Stamp / Signature of the Invigilator

EXAMINATION (Mid Semester)

SEMESTER (Spring)

Roll Number

Section

Name

Subject Number

C S 1 0 0 0 3

Subject Name

Programming and Data Structures

Department / Center of the Student

Additional sheets

Important Instructions and Guidelines for Students

1. You must occupy your seat as per the Examination Schedule/Sitting Plan.
2. Do not keep mobile phones or any similar electronic gadgets with you even in the switched off mode.
3. Loose papers, class notes, books or any such materials must not be in your possession, even if they are irrelevant to the subject you are taking examination.
4. Data book, codes, graph papers, relevant standard tables/charts or any other materials are allowed only when instructed by the paper-setter.
5. Use of instrument box, pencil box and non-programmable calculator is allowed during the examination. However, exchange of these items or any other papers (including question papers) is not permitted.
6. Write on both sides of the answer script and do not tear off any page. **Use last page(s) of the answer script for rough work.** Report to the invigilator if the answer script has torn or distorted page(s).
7. It is your responsibility to ensure that you have signed the Attendance Sheet. Keep your Admit Card/Identity Card on the desk for checking by the invigilator.
8. You may leave the examination hall for wash room or for drinking water for a very short period. Record your absence from the Examination Hall in the register provided. Smoking and the consumption of any kind of beverages are strictly prohibited inside the Examination Hall.
9. Do not leave the Examination Hall without submitting your answer script to the invigilator. **In any case, you are not allowed to take away the answer script with you.** After the completion of the examination, do not leave the seat until the invigilators collect all the answer scripts.
10. During the examination, either inside or outside the Examination Hall, gathering information from any kind of sources or exchanging information with others or any such attempt will be treated as '**unfair means**'. Do not adopt unfair means and do not indulge in unseemly behavior.

Violation of any of the above instructions may lead to severe punishment.

Signature of the Student

To be filled in by the examiner

Question Number	1	2	3	4	5	6	7	8	9	10	Total
Marks Obtained											
Marks obtained (in words)				Signature of the Examiner				Signature of the Scrutineer			

Instructions to students

- Write your answers in the question paper itself. Be brief and precise. Answer all questions.
 - Write the answers only in the respective spaces provided.
 - The questions appear on the next six pages.
 - Space is provided for doing rough work. Please do not write the solutions in the rough spaces. Ask for supplementary sheets during the test if you need more rough space.
 - All the questions use the programming language C.
 - Not all blanks carry equal marks (unless otherwise stated). In those cases, evaluation will depend on overall correctness.
-

Do not write anything on this page.

1. Answer the following parts by filling in the blanks. Write only the answers. No need for any explanation. (2 × 5)

(a) What will be the output of the following code segment?

```
int a = 30, b = 12, c, d;
float x, y;
c = a / b * b;
d = a++ + --c;
x = (float) a / b * b;
y = (float) (a / b) * b;
printf ("%d %d %f %f", c, d, x, y);
```

Answer: 23 53 31.000000 24.000000

(b) What will be the output of the following code segment?

```
int a = 10, b = 20, c;
if (a > b) c = a + b;
else if (a % 5 == 0) c = a * 5;
else c = a - b;
printf ("%d", c);
```

Answer: 50

(c) What will be the output of the following code segment?

```
int sum = 0, count = 0;
while (count < 20) {
    sum += count--;
    count += 3;
}
printf ("%d %d", count, sum);
```

Answer: 20 90

(d) What will be the output of the following code segment?

```
int a = 119, b = 34;
do {
    if (a > b) a = a - b;
    else b = b - a;
} while (a != b);
printf ("%d", a);
```

Answer: 17

(e) What will be the value of **s** immediately after the following code segment is executed?

```
int s, i;
for (s = 0, i = 1; i <= 6; s = s + i * i, i++);
```

Answer: 91

Rough Work

2. (a) Consider the following code snippet.

```
int i;
scanf("%d", &i);
if (i == 1)
    printf("i = %d\n", i);
else if (i >= -2 && i <= 2)
    printf("i = 1\n");
else if ((i > 4 || i < -4) && (i == 5 || (i += 1)))
    printf("i = 5\n");
else if (i = 4)
    printf("i = %d\n", i);
```

Fill in the blanks in the code snippet below such that for all values of *i*, the switch statement below produces exactly the same output as the above snippet. (4)

```
int i;
scanf("%d", &i);
switch (i)
{
    case _____ -2 _____ :
    case _____ -1 _____ :
    case _____ 0 _____ :
    case _____ 1 _____ :
    case _____ 2 _____ : printf("i = 1\n"); break;
    case _____ -4 _____ :
    case _____ -3 _____ :
    case _____ 3 _____ :
    case _____ 4 _____ : printf("i = 4\n"); break;
    default: printf( _____ "i = 5\n" _____ );
}
```

Rough work

(b) In the following three code snippets, **S** is a single statement and is the same in all the three snippets.

```
Snippet A
int i;
for (i=0; i<10; i++) {
    if (i % 2 == 0)
        S
    else
        i++;
}
printf("i = %d\n", i);
```

```
Snippet B
int i = 0;
while (i<10) {
    if (i % 2 == 0)
        S
    else
        i++;
    i++;
}
printf("i = %d\n", i);
```

```
Snippet C
int i = 0;
while (i<10) {
    i++;
    if (i % 2 == 0)
        S
    else
        i++;
}
printf("i = %d\n", i);
```

Answer the following questions in relation to these three snippets. Fill in the blanks provided. No need for any explanation.

(i) Suppose that **S** is the statement $i = a * i + 1$; where **a** is a positive integer constant. What is the smallest value of **a** such that not all of the three snippets produce the same output? 2 (2)

(ii) Suppose that **S** is the statement $i = b$; where **b** is an integer constant. What is the largest value of **b** such that exactly one of the three snippets will have an output? 7 (3)

(iii) Suppose that **S** is the statement $i = c$; where **c** is an integer constant. What is the unique value of **c** such that all the three snippets have the same output? 9 (3)

Rough work

3. (a) For the following loop, write your answers in the blanks.

```
for (k=1; k<100; ++k)
    k *= (k + 1);
```

How many times is the statement `k *= (k + 1);` executed? 3 (1)

How many times is the continuation condition `k<100` checked in the loop? 4 (1)

What is the value stored in the variable `k` immediately after the `for` loop terminates? 183 (2)

(b) What will be the outputs of the following programs? Write in the blank space below each program. (2 × 4)

```
#include <stdio.h>
int main()
{
    int s = 0;
    while (s++ < 10) {
        if ((s < 4) && (s < 9)) continue;
        printf("%d ", s);
    }
    return 0;
}
```

4 5 6 7 8 9 10

```
#include <stdio.h>
int main()
{
    int x = 4, y = 6, z = 0;
    while (y != 0) {
        if (y % 2) z += x;
        x *= 2;
        y /= 2;
    }
    printf("%d", z);
    return 0;
}
```

24

```
#include <stdio.h>
int main()
    int x = 0, y = 10, z = 20;
    while (1) {
        x++;
        if (y > z) break;
        y += 4*x; z += 2*x;
    }
    printf("x = %d, y = %d, z = %d", x, y, z);
    return 0;
}
```

x = 4, y = 34, z = 32

```
#include <stdio.h>
int main()
    int p, q;
    for (p=q=0; p<10; ++p) {
        q = p + q;
        ++p;
    }
    printf("%d", q);
    return 0;
}
```

20

Note: There is a typo in the bottom two programs of Q3(b). The opening braces after `main()` are missing. If you fail to notice the typos (or ignore them) and proceed with the intended meanings of the programs, your answers will be evaluated according to the correctness. If you notice the typo and answer *error* or *no output* or some such similar thing, you will get full credit.

Rough work

4. (a) In this part of this exercise, we use the infinite series

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$$

$$\log(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots$$

for computing $\sin(x)$ and $\log(1+x)$. Assume that x is a positive real number in the range $-1 < x < 1$ (otherwise the log series is not valid). The user enters x (in this range) and the number n of terms of the series used in the computation. A single loop computes both $\sin(x)$ and $\log(1+x)$. Fill in the blanks to complete the following program for this purpose. (8)

```
#include <stdio.h>
int main()
{
    int n;
    double x, sin_result=0, sin_term, log_result=0, log_term;
    scanf("%lf%d", &x, &n);

    sin_term = _____ x _____ ; /* first term of sin(x) */

    log_term = _____ x _____ ; /* first term of log(1+x) */
    for (int i = 1; i <= n; i++) {
        sin_result += sin_term;
        log_result += log_term;

        sin_term *= _____ -x * x / ((2 * i) * (2*i + 1)) _____ ;

        log_term *= _____ -x * i / (i + 1) _____ ;
    }
    printf("sin(%lf) = %lf, log(%lf) = %lf",
        _____ x, sin_result, 1+x, log_result _____ );
    return 0;
}
```

(b) Fill in the blanks so that the following C program reads a and b from the user, correctly computes the product $(a+b)(a-b)$, and prints this product up to two decimal places. (4)

```
#include <stdio.h>

#define mult(x,y) _____ (x) * (y) _____

int main()
{
    int a;
    double b;

    scanf(" _____ %d _____ ", &a);

    scanf(" _____ %lf _____ ", &b);

    printf(" _____ %.2lf _____ ", mult(a+b,a-b));
    return 0;
}
```

5. For a positive integer n , let $\text{half}_1(n)$ denote the integer obtained by taking every alternate digit of n starting from the left end. The digits not included in $\text{half}_1(n)$ constitute the other half called $\text{half}_2(n)$.

Examples

n	$\text{half}_1(n)$	$\text{half}_2(n)$
2345678	2468	357
23456789	2468	3579
4050607	4567	0
44556677	4567	4567

- (a) A function `half(n, which)` takes an integer `n` and a flag `which` as its only arguments. If `which = 1`, the function returns $\text{half}_1(n)$. If `which = 2`, the function returns $\text{half}_2(n)$. For all other values of `which`, the function returns 0. The digits of n are processed from the left end of n . If l is the number of decimal digits in n (without leading 0's), then we have $10^{l-1} \leq n < 10^l$. The function starts by computing $t = 10^l$ (the value of l is not needed). Then, the values $10^{l-1}, 10^{l-2}, \dots, 10^1, 10^0$ are used in that sequence to extract the digits of n one by one. The loop computes $\text{half}_1(n)$ in `h1` and $\text{half}_2(n)$ in `h2`. The digits of n are accumulated alternately in `h1` and `h2`. This is governed by the flag `turn`. If `turn = 1`, then the digit goes to `h1`. If `turn = 2`, then the digit goes to `h2`. After the loop ends, the input parameter `which` is looked at, and the appropriate value (`h1` or `h2` or 0) is returned. Fill in the blanks to complete the code of the function `half()`. Do not use any variables/arrays other than those given, or any logic other than what is mentioned above. (12)

```

_____ int _____ half ( _____ int n, int which _____ )
{
    int d, t;
    int h1, h2, turn;

    t = 1;
    /* Compute  $t = 10^l$  in a loop. No need to compute the value of  $l$ . */
    _____ while (n >= t) t *= 10;

    t /= 10; _____ /* We need  $10^{l-1}, 10^{l-2}, \dots, 10^1, 10^0$  in the loop below */
    h1 = h2 = 0; turn = 1; /* Initialize variables */

    /* Repeat the following loop until all digits of n are extracted */
    while ( _____ t > 0 _____ ) {
        d = _____ n / t _____ ; /* Extract the next digit from n */
        n = _____ n % t _____ ; /* Remove this digit from n */
        t /= 10; _____ /* Update power of 10 */
        /* Depending on turn, update h1 or h2 */

        _____ if (turn == 1) h1 = 10 * h1 + d; else h2 = 10 * h2 + d;

        turn = _____ 3 - turn _____ ; /* Flip the value of turn */
    }

    _____ if (which == 1) return h1;
    _____ if (which == 2) return h2;
    _____ return 0;
}

```

- (b) Define two macros for computing $\text{half}_1(n)$ and $\text{half}_2(n)$. (2)

```

#define half1(n) _____ half(n, 1)
#define half2(n) _____ half(n, 2)

```