

# CS11002 Programming and Data Structures, Spring 2008

## Class test 1

Total points: 20

February 05, 2008

Total time: 1 hour

Roll no: \_\_\_\_\_ Name: \_\_\_\_\_ Section: \_\_\_\_\_

[ Write your answers in the question paper itself. Be brief and precise. Answer all questions. ]

1. (a) Which one of the following is a valid name of a C variable: `2ab_c`, `switch`, `xy#1`, `"rst"`? (f) How many times is the statement `i *= i+1;` (1×10) in the following `for` loop executed?

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

Ans: `switch`

- (b) The `printf()` function returns the number of characters it prints on `stdout` (screen). What value will be stored in `count` after the execution of the following code?

```
int count, n = 100;
count = printf("\nn:%d\n",n);
```

Ans: 7

- (c) What will be the contents of the variables `a` and `b` after the execution of the following code on the input `I do not know`?

```
char a, b;
scanf("%c do not %c", &a, &b);
```

Ans: The characters `'I'` and `'k'`, respectively

- (d) If the number of bits in the memory address of a computer is 16, what is the maximum number of addressable memory locations?

Ans:  $2^{16} = 65536$

- (e) What values does the following code print?

```
int m,n;
m = n = 4;
m *= 3/2;
n = n * 3/2;
printf("%d %d", m, n);
```

Ans: 4 6

Ans: Three times

- (g) How many times is the loop condition `i<100` checked in the loop of Part (f)?

Ans: Four times

- (h) What is the value stored in the variable `i` immediately after the loop of Part (f) terminates?

Ans: 183

- (i) What is printed by the following code?

```
int a = 4, b = 6, c = 4;
if (a > b < c) printf("A");
else if (a > b) printf("B");
else if (b < c) printf("C");
else printf("D");
```

Ans: A

- (j) What value does the following code print?

```
#define N a*b
int a = 5, b = 10, c = 15;
printf("%d",c/N);
```

Ans: 30

2. In the following C code segment, **p**, **x** and **y** are **unsigned int** variables. The code segment computes a function  $f(x, y)$  in the variable **p**. Determine  $f(x, y)$ .

```
p = 0;
while (y != 0) {
    if (y % 2) p += x;
    x *= 2; y /= 2;
}
```

Ans:  $f(x, y) = xy$

(4)

3. For a real number  $x$ , the notation  $\lfloor x \rfloor$  stands for the largest integer less than or equal to  $x$ . For example,  $\lfloor \pi \rfloor = 3$  and  $\lfloor 3 \rfloor = 3$ . You are to write a program that reads a positive integer  $n$  and an integral base  $b \geq 2$ . The program computes and prints the value of  $\lfloor \log_b n \rfloor$ . For example,  $\log_{23} 456789 = 4.1562752022\dots$  and so  $\lfloor \log_{23} 456789 \rfloor = 4$ . Therefore, upon input  $n = 456789$  and  $b = 23$ , your program should print 4.

Complete the following C program so as to achieve this goal. You are **not allowed** to use any math library call (like **log**, **log10** or **floor**). Do not make any floating point calculations. Do not write any function (other than **main**). You may, however, declare and use some additional **int** variables (but no arrays). (6)

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

```
int main ()
{
```

```
    int n,b,t,m; /* An additional variable m is declared here */
```

```
    printf("Enter a positive integer    : "); scanf("%d", &n);
```

```
    printf("Enter an integer base >= 2 : "); scanf("%d", &b);
```

```
    /* Now complete the code for computing  $\lfloor \log_b n \rfloor$  */
```

```
    t = 0;
```

```
    m = n; /* We should not destroy n, since it will be printed at the end */
```

```
    while (m >= b) {
```

```
        m /= b;
```

```
        ++t;
```

```
    }
```

```
    printf("The integer logarithm of %d to base %d is %d\n", n, b, t);
```

```
}
```