CS11001 Programming and Data Structures, Autumn 2010

Class Test 1

Maximum marks: 20   September 06, 2010   Total time: 1 hour

Roll no: _____   **Name:** _____   **Section:** _____

$\left[\text{\textit{Write your answers in the question paper itself. Be brief and precise. Answer \underline{all} questions.}}\right]$

1. (a) What does the following program print? **(2)**

```c
#include <stdio.h>

main ()
{
    int i = 100, m, n;

    m = 1 + (i++);
    n = 1 + (++i);
    printf("m = %d, n = %d\n",m,n);
}
```

(b) What does the following program print? **(3)**

```c
#include <stdio.h>

int eval ( int q )
{
    q *= 10;
    q -= 100;
    return q;
}

main ()
{
    int p = 10, q = 100;

    q = eval(p);
    printf("%d %d ", p, q);
    q = eval(q);
    printf("%d %d\n", p, q);
}
```

(c) Describe in one English sentence what the following function outputs. Assume $b \neq 0$. **(2)**

```c
int what ( int a, int b )
{
    double q1, q2;

    q1 = a / b;
    q2 = (double)a / (double)b;
    return (q1 == q2);
}
```

(d) Express $f(n)$ as a function of n, where f is defined as follows. Show your calculations. **(3)**

```c
unsigned int f ( unsigned int n )
{
    unsigned int s = 0, i, j;

    for (i=1; i<=n; ++i)
        for (j=i; j<=n; ++j)
            s += i + j;
    return s;
}
```

2. In this exercise, we compute the binomial coefficient $\binom{n}{r}$ by repeatedly using the formula $\binom{n}{r} = \frac{n}{r}\binom{n-1}{r-1}$. We compute n/r as a floating-point value. Finally, the accumulated product is rounded to the nearest integer. In both the following parts, you are <u>not allowed to use any math library function</u>.

(a) Fill in the blanks to complete the following C function that takes a floating-point value x as its only argument and returns the rounded value of x. The rounded value of x is the integer nearest to x. When x is mid-way between two consecutive integers, we follow the convention "round half away from zero", that is, round$(2.5) = 3$ and round$(-2.5) = -3$. **(4)**

```
int roundit ( double x )
{
    int r;          /* The rounded integer to return */
    double fpart;   /* Fractional part */

    /* Store in r the truncated value of |x| */

    _____

    /* Store in fpart the fractional part of |x| */

    _____

    /* Modify r based conditionally upon fpart */

    if (fpart >= 0.5) _____ ;

    /* Return r after sign adjustment */

    return _____ ;
}
```

(b) Complete the following C function to compute $\binom{n}{r}$. The function starts by initializing an empty product, and subsequently multiplies the product by $\frac{n}{r}, \frac{n-1}{r-1}, \frac{n-2}{r-2}$, and so on, in a loop, until the denominator reduces to 0. After the loop terminates, the product is almost ready to be returned (since $\binom{n-r}{0} = 1$). But since floating-point calculations are used to compute the product, there may be some (small) error due to floating-point approximations. Use the function of Part (a) to round the product, and return the rounded value. **(6)**

```
unsigned int bincoeff ( unsigned int n, unsigned int r )
{
    double prod = _____ ;   /* Initialize to empty product */

    if (r > n) return _____ ;

    while ( _____ ) /* Condition on r */ {
        /* Multiply prod by the fraction n/r (floating-point division) */

        _____

        /* Prepare for the next iteration (computation of (n-1 choose r-1)) */

        _____
    }

    /* Return the rounded product. Use the function of Part (a). */

    return _____ ;
}
```