

MA U11601 Quiz 03 12/11/21 ANSWERS

Answer any 3 questions. Submit them through Blackboard as pdfs, either handwritten and scanned, or typeset. They should be submitted before midnight on Wednesday 17 November. If more than three answers are submitted, only the first three will be marked. All questions carry 20 marks.

Show all work. That is, where an answer requires some calculation, show the calculation.

Plagiarism. If copying is detected, all those involved will lose credit, irrespective of who copied from whom.

Question 1. Given two short ints 0c45, 9841 in hexadecimal (big endian), (i) Compute their sum as a short int; (ii) convert all three short ints to decimal.

Answer.

```
% a.out 3141 -26559
m 3141: 0c 45
n -26559: 98 41
m+n -23418: a4 86
```

Question 2. Write an *efficient recursive* routine `double power ( double a, int n )` which assumes $n \geq 0$ and returns a^n . It should be modelled on ‘Russian peasant multiplication’ in section 13.5 of the notes. By ‘efficient’ is meant that it uses $2 \log_2(n)$ multiplications, roughly, instead of n if one uses a for-loop in the usual way.

Answer.

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

double power ( double a, int n ) // assumed nonnegative
{ if ( n == 0 )
    return 1;
  else
  { int p = power ( a, n/2 );
    if ( n%2 == 0 )
      { return p * p; }
    else
      { return p * p * a; }
  }
}

// only the function is required, but this illustrates it
int main ( int argc, char * argv[] )
{
  double a = atof ( argv[1] ); int n = atoi ( argv[2] );
  printf("%f ^ %d is %f\n", a, n, power(a,n));
}
```

Question 3. Write a function

```
int is_leapyear ( int yy )
```

which returns 1/0 according as yy (which may be assumed to be a 4-digit decimal number ≥ 1582) is or is not a leap year.

Remember that every multiple of 4 is a leap year, with the exception of centuries: every 4th century is a leap year, the other centuries are not.

Answer.

```
int is_leapyear ( int yy )
{
    if (yy % 4 != 0)
        return 0;
    else if (yy % 400 == 0)
        return 1;
    else if (yy % 100 == 0)
        return 0;
    else
        return 1;
}
```

Question 4. Simulate xxx(3,5) given

```
int xxx ( int m, int n )
{
    int i, xx;
    xx = 1;
    for ( i=0; i<n; ++i )
    { xx *= m; }
    return xx;
}
```

In Section 12.11, a function which contains a for-loop, is simulated. Your answer should be similar.

Answer.

m	n	xx	i	i<n
3	5			
		1		
			0	
				y
		3		
			1	
				y
		9		
			2	
				y

```

        27
          3
            y
      81
        4
          y
    243
      5
        n
return 243

```

Question 5. Write a *recursive* function

```
int lesseq_count ( double x, int n, double a[] )
```

which returns the number of entries $a[i]$, $0 \leq i < n$, such that $a[i] \leq x$. (Assume $n \geq 0$.)

Answer.

```

int lesseq_count ( double x, int n, double a[] )
{
    if ( n == 0 )
        return 0;
    else if ( a[n-1] <= x )
        return 1 + lesseq_count ( x, n-1, a );
    else
        return lesseq_count ( x, n-1, a );
}

```