

CS 007A – Midterm 1 Practice – Chapters 1—5

1. Consider the 4 bytes stored in contiguous memory, as shown to the right.
 - a. Determine the decimal representation of these 4 one-byte integers:
 - b. Determine the decimal form of these, interpreting the data as 2 two-byte integers are stored in these 4 bytes.
 - c. Interpret the 4 bytes as 4 ASCII characters. Note that the ASCII value of A is 65.

⋮
01000001
01010011
01001011
01011001
⋮

2. Write 125 in binary form.
3. Write 125 in Hex form.
4. Convert the hex number abba to decimal form.
5. Find as many errors in the following code as you can. Classify each error as a syntax error or a logic error and describe how to fix it so it works properly.

```
//This program uses a loop to raise a number to a power.
#include <iostream>
using namespace std;

int main()
{ int num, bigNum, power, count;
  cout << "Enter an integer: ";
  cin >> num;
  cout << "\nWhat power do you want it raised to? ";
  cin >> power;
  bigNum = num;
  while (count++ < power);
    bigNum *= num;
  cout << "\nThe result is " << bigNum << endl;
  return 0;
}
```

6. Consider the following C++ program. Is there an error? If so what kind of error? What will be the result of attempting to compile and execute the program?

```
#include <iostream>
using namespace std;

int main()
{ for(int i = 0; i < 2; ++i)
  for(int j = 0; j < 2; ++ j)
  {
    cout << "i = " << i << " ";
    --i;
    cin.get();
  }
  return 0;
}
```

7. Evaluate the following expressions, step by step, assuming integer variables a = 52, b = 13, c = 4, d = 5 and float e = 2.71828. If the expression doesn't parse, explain why.
 - a. a / d % b * c
 - b. a % c == b

- c. $!(b + d != c * c)$
- d. $d \% b * c > 5 \ || \ c \% b * d < 7$
- e. $((a + 1) * b + 2) \% c + 3 / d + e$

8. What is the output of the following code fragment?

```

for (int ct = 1; ct <= 3; ct++) {
    cout << ct;
    for (int i = 3; i <= 4; i++) {
        cout << ct << " " << i;
    }
    cout << "do re mi";
}

```

9. What is the output produced by the following program if the user enters 4?

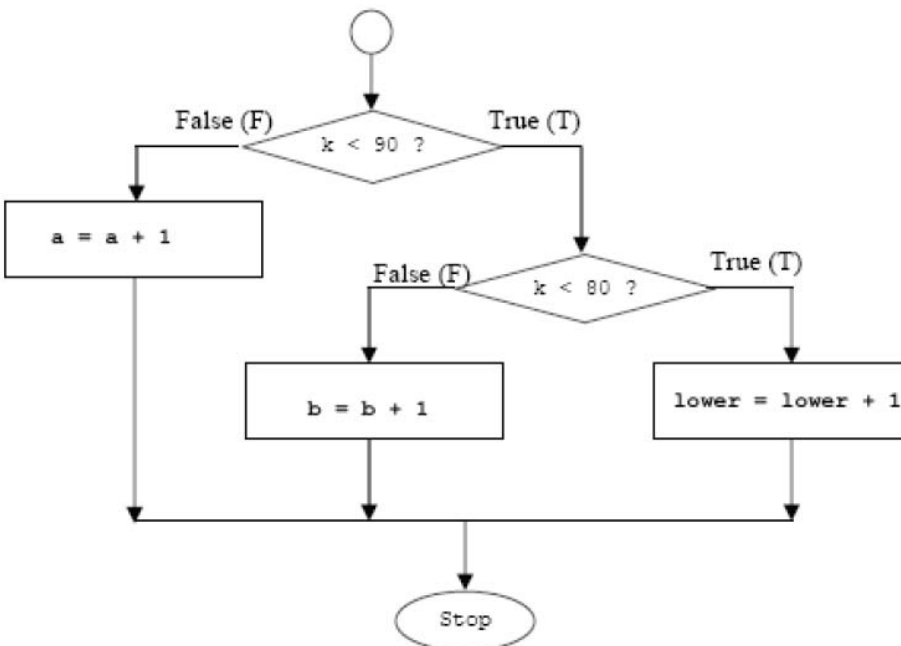
```

#include <iomanip>
#include <iostream>
using namespace std;

int main()
{
    setiosflags(ios::right);
    int n;
    cout << "How many columns? (1-16): ";
    cin >> n;
    for (int x=1; x <= n; x++)
    {
        for (int y=1; y <= n; y++)
            cout << setw(5) << x*y;
        cout << endl;
    }
}

```

10. Write code to implement the following flow chart. The user will enter 10 numbers between 0 and 100 and the program will count how many are ≤ 90 , how many are in the interval $[80, 90)$ and how many are less than 80 and then report the results.



11. Write a program to

- Find the square root of the sum of the first 100 even numbers, correct to 10 significant digits. You may use not the `cmath` library.
- Find the reciprocal of the product of the first 10 odd numbers correct to 10 significant digits.
- Fill up an array of 3 elements with 3 random numbers between 1 and 10 and then compute the cube root of their product. Do not use the `cmath` library.
- Compute the value of the zeta function, $\zeta(s) = \frac{1}{1^s} + \frac{1}{2^s} + \frac{1}{3^s} + \dots$ for any positive integer $s > 1$.

12. Consider the following program fragment.

```
int accumulator = 0, sam, pam;
cout << "\nEnter integers for sam and pam: ";
cin >> sam >> pam;
while (true )
{
    if (pam == 0) break ;
    accumulator += ((pam % 2 == 1) ? sam : 0);
    pam /= 2;
    sam *= 2;
}
cout << accumulator << "\n";
```

a. Complete the following tables until the program completes, or indicate that it's an infinite loop:

sam	pam	accumulator
5	4	0
⋮	⋮	⋮

sam	pam	accumulator
6	17	0
⋮	⋮	⋮

b. In a few words, describe the output of this program in terms of the input values of `sam` and `pam`, and how it works.

13. Suppose that `p = 0x0012FF50`, `&a = 0x0012FF44`, `&p = 0x0012FF38`.

What will be output by the following code?

```
int main()
{ int a[] = { 22, 33, 44, 55, 66, 77, 88, 99 };
  int* p = &a[3]; // p points to a[3]
  cout << "p = " << p << ", *p = " << *p;
  cout << "\n&a = " << &a; // ok: a is an lvalue
  cout << "\n&p = " << &p; // ok: p is an lvalue
  cout << "\n&(a[5]) = " << &(a[5]); // ok: a[5] is an lvalue
  cout << "\n&(*(a+5)) = " << &(*(a+5)); // ok: *(a+5) is an lvalue
  cout << "\n&*(p+2) = " << &*(p+2); // ok: *(p+2) is an lvalue
  cout << "\n&(p+2) = " << p+2; //&(p+2); // ERROR: p+2 is not an lvalue
  cout << endl;
}
```

14. Write a C++ program to implement the flow chart below.

