CSCI 13500 midterm f21 v2 (yellow)

TOTAL POINTS

94 / 100

QUESTION 1

30 pts

1.1 3 / 3

✓ **- 0 pts Correct**

- 3 pts wrong answer

1.2 3 / 3

✓ **- 0 pts Correct**

- 3 pts Incorrect

1.3 0 / 3

- 0 pts Correct

✓ **- 3 pts Incorrect**

1.4 3 / 3

✓ **- 0 pts Correct**

- 3 pts Incorrect

1.5 3 / 3

✓ **- 0 pts Correct**

- 3 pts Incorrect

1.6 3 / 3

✓ **- 0 pts Correct**

- 3 pts Incorrect

1.7 3 / 3

✓ **- 0 pts Correct**

- 3 pts Incorrect

1.8 3 / 3

✓ **- 0 pts Correct**

- 3 pts Incorrect

1.9 3 / 3

✓ **- 0 pts Correct**

- 3 pts Click here to replace this description.

1.10 3 / 3

✓ **- 0 pts Correct**

- 3 pts Incorrect

Don't forget to declare an int!

QUESTION 2

30 pts

2.1 10 / 10

✓ **+ 10 pts Correct**

+ 5 pts Partly

+ 2 pts Incorrect

+ 0 pts Blank

2.2 10 / 10

✓ **+ 2.5 pts** a = 3

✓ **+ 2.5 pts** b = 6

✓ **+ 2.5 pts** c = 2

✓ **+ 2.5 pts** d = 8

+ 1 pts Incorrect

+ 0 pts Blank

2.3 10 / 10

✓ **- 0 pts Correct**

- 3 pts foo(8) is not a number, it is string "10"

- 3 pts foo(17) is not a number, it is string "21".

- 2 pts no intermediate step and results is wrong.

- 2 pts foo(8) is not the string in your paper.

- 2 pts foo(17) is not the string in your paper.

- 1 pts calculation is not correct.

- 9 pts no actual output

- 3 pts You reverse the string.

- 3 pts to_string(2) is "2". You need to know

concatenation of string.

- 0.5 pts foo(17) is a string, which should be included in "".

- 1 pts output is string and should be included in "".

QUESTION 3

3 20 / 20

✓ **+ 20 pts Correct**

+ 5 pts correct function declaration

+ 5 pts correct for loop or for - if

+ 5 pts string build up (e.g. ans_str += str[i] or ans_str = ans_str + str[i])

+ 5 pts correct return

+ 2 pts Write something

QUESTION 4

4 17 / 20

- 0 pts Correct

✓ **- 1 pts When a is 0, if b is also 0, x can be any value, but if a is 0 but b is not 0, then there is no solution.**

- 0.5 pts Expression b x b should be b * b.

- 0.5 pts Type of x is double, not int.

- 1 pts a = 0 b = 0 is not a valid condition. You mean (a == 0 && b == 0)?

- 1 pts Need to use curly parentheses to enclose statements in if- or else- part if there is more than one statement in those parts.

- 1 pts Need to print out result.

- 0.5 pts a is not initialized.

- 1 pts x is not declared.

- 1 pts output x as -(b*b)/a is when a is not 0. That is, you need to enclose this statement in else part of if (a == 0).

✓ **- 0.5 pts Should not use . to replace multiplication operator ***

- 1 pts solution when a is not zero is not correct. It should be -(b*b/a);

✓ **- 1.5 pts You change problem. We did not require that a cannot be zero.**

- 1 pts print result, not return its value in main function, which can only return an int.

- 0.5 pts a and b are double, not int. The result is a double type.

- 0.5 pts int a; b; x; should not use ; You either use double a; double b; double x; OR you can use double a, b, x;

- 1 pts condition (a = 0) should be (a == 0)

- 2 pts initialize a and b.

- 0.5 pts x is -(b*b) / a;

- 2 pts Do not solve x correctly when a is not zero.

- 1 pts x is not entered, it is calculated from a and b.

- 4 pts calculation is not correct.

- 1 pts Why use for (int i = 0; i <= 0; i++) return [i]; [i] is not correct in syntax.

- 0.5 pts no declaration of formula

- 0 pts break; normally only works in repetition statement and switch statement.

- 1 pts x calculation is not correct

- 1 pts Formula to calculate for x does not fit for C++ syntax.

- 0.5 pts x is only calculated when a is not zero, so print its value only in the else-part. You cannot put it outside the if-else statement.

- 0.5 pts result should be -b*b / a; or -pow(b, 2)/ a; You miss negative sign.

- 0.5 pts cannot put = between type and variable names. So int = a, b, x; is wrong.

- 0.5 pts There is no else if in C++. You either use else if (condition) or else.

- 0.5 pts A statement needs to be ended by semicolon ;

- 0.5 pts pow(-1*b, 2) or pow(-b, 2) is the same as pow(b, 2). The answer for x is -pow(b, 2) / a; Negative sign should be out of pow function.

- 0 pts main function header should be int main()

- 20 pts no submission

- 0 pts Need to enter values from console. Do not initialize them directly.

- 0.5 pts Only need to enter values for a, b. No need to enter values for other variables.

- 5.5 pts We are not solving for quadratic equation since there is no x^2 in the equation.

- **1 pts** 2a should be 2 * a, and 2 * a should be enclosed in ().
- **0.5 pts** 4(ac) should be 4 * a * c.
- **0 pts** cout << "result2"; print out string result2. cout << result2; print out the value of result2.
- **1 pts** cannot have two unconditional return statements. For example, return x; followed by return 0; then the second return statement will not run.
- **1 pts** x can only be calculated when a is not 0. So the calculation of x and print its result should be in else part of if (a == 0) statement.
- **1 pts** print x after calculate it.
- **0.5 pts** b^2 is written as b * b
- **2 pts** condition in while statement is wrong. In fact, we do not need repetition statement in this problem.
- **9 pts** calculation is not correct. Not close.
- **1 pts** No declaration and calculation of x.
- **0.5 pts** Calculation of x only happens when a is not zero. You either put the calculation and print x in else part of if (a == 0) or put a return statement in if-part.
- **1 pts** No declaration of variable result.
- **3 pts** Need an if statement to handle the case when a is zero.
- **0.5 pts** No input for b.
- **0.5 pts** num is not int, it is double type.
- **0 pts** Please write bigger font.
- **0.5 pts** calculate -b*b/a should be put inside the if-part of if (a == 0), otherwise, exception divided by 0 will happen.
- **1 pts** pow * equation = a x + b . b is not a valid statement.
- **1 pts** no declaration of missingvalue.
- **1 pts** Why do you need to use enum?
- **1 pts** cin > "a"; should be cin >> a; cin >> "b" should be cin >> b;
- **1 pts** If you define foo function, you need to how to call it in main function.
- **1 pts** double(double x) and a*x + pow(b,2) = 0 cannot be compiled.
- **9 pts** calculation for x when a is not 0 is missing.
- **1 pts** cout "x = 0" is not correct. You mean cout << "x = 0"; But the logic is not correct. When a is 0 and b is 0, variable x can be any value.
- **0 pts** int a, b; x; should be double a, b, x;
- **1 pts** Need to put the calculation of x in the else part of if (a == 0).
- **0.5 pts** condition needs to be put in a pair of parentheses in C++
- **0.5 pts** result -= 0.0; does not change result since it is equivalent to result = result - 0.0; you mean result = 0 - result? Note that result = 0 - result; cannot be simplified as result -= 0;
- **1 pts** do-while loop can be taken out, since a can be zero.
- **1 pts** double ans = 0; should be outside if-else statement, otherwise, if double ans = 0; only in if-part, ans is not visible to else-part.
- **1 pts** a is not zero is written as (a != 0) or (!(a == 0)).
- **1 pts** We did not exclude the case when a is 0. So you should not force the user enter a value that a is not zero.
- **0.5 pts** Between -1 and pow(b, 2), there should be multiplication operator *
- **0.5 pts** No declaration of x
- **2 pts** When a is 0, need to consider the case when b is zero or not.
- **2 pts** logic is not correct. we only need to consider the case when a is zero and a is not zero.
- **0.5 pts** When a and b are both 0, x can be any number.
- **1 pts** cin >> a; cin >> b; or cin >> a >> b;
- **1 pts** float a == 0 is not a valid condition.
- **1 pts** Misrepresent pow(b, 2) with sqrt(b)
- **0.5 pts** condition needs to be enclosed in a pair of parentheses
- **3 pts** When a is zero, no actual processing.
- **0.5 pts** In main function, return an int, not just return;
- **1 pts** Need to put code to calculate and print x when x is not zero inside main function. Every piece code need to be put inside a function.
- **0.5 pts** -pow(b, 2) % a should be -pow(b, 2) / a;
- **5 pts** no code to handle a is zero.

- **1 pts** no declaration of term2
- **0.5 pts** -b^2 is not valid in C++
- **2 pts** In main function, cannot return a double or a boolean.
- **1 pts** Only calculate x when a is not zero.
- **1 pts** condition a > 0 should be a != 0
- **1 pts** pow(2, b) is not the same as pow(b, 2).
- **3 pts** function is not correct. Return type is double, not int, and you did not return anything when a is zero.
- **3 pts** Need to write code in main function to call function defined.
- **0.5 pts** b^2 is b * b in C++
- **0.5 pts** need to write s / a in C++
- **1 pts** do is not a complete statement, you means else?

1. (30 points) Answer the following questions.

(1) What is the value of 7 / 2 * r * r when variable r is 2 in C++?

$7/2=3 \quad 3 \cdot r \cdot r = 3r^2 = 12$

$3 \cdot 2 \cdot 2 = 12$

(2) Declare function foo whose input parameter is a string and return is an int. You just need to write the function header, no implementation is needed.

```
int foo(string p);
```

(3) Write code to generate a random int in [-100, 100].

$(int)\left(\frac{rand()}{RAND\_MAX} \cdot 201\right) - 100$

(4) Given array of strings as follows
string greetings[] = {"Hello", "Morning", "Hi", "Great"};
What is the value for greetings[3].length()?

5

(5) Suppose we generate a runnable file myprog, and we would like redirect the input from console to a file called input.txt. What is the command?

```
./myprog < input.txt
```

value = 0

(6) What is the output of the following code?

```
int value = 0;
for (int i = 1; i < 6; i += 2)
    value += i;

cout << value;
```

i=1, value=1
i=3, value=4
i=5, value=9
i=7, done

9

(7) Write code to declare an array of int with size 60, call it **scores**. Initialize each element by 1.

```
int scores[60] = {3;
for (int i=0; i<60; i++)
    scores[i] = 1;
```

(8) What is the output of the following code?

```
for (int i = 0; i < 4; i++)
{
    for (int j = 0; j < 4; j++)
        if (i % 2 != j % 2)
            cout << "X";
        else cout << "O";

    cout << endl;
}
```

OXOX
XOXO
OXOX
XOXO

(9) Write a condition to represent that both x and y are out of the range of [0, 100], where both ends are included. Suppose x and y are properly declared and initialized.

(x<0 || x>100) && (y<0 || y>100)

(This is AND)

(10) Suppose n is an int, write code to get its last digit. For example, suppose n is 21, after your code, you get 1.

n % 10

2. (30 points) Short answer questions

(2.1) Given three integers a, b and c, properly declared and initialized, write a code to find out the minimum number.

```
int min = std::min(a, std::min(b,c));
```

(2.2) Read codes and write output.

```
void foo(int& a, int& b);

int main()
{
   int a = 3;
   int b = 6;

   foo(a, b);
   cout << "a = " << a << endl;
   cout << "b = " << b << endl;

   int c = 8;
   int d = 2;

   foo(c, d);
   cout << "c = " << c << endl;
   cout << "d = " << d << endl;

   return 0;
}

void foo(int& a, int& b)
{
   int temp;
   if (a % b == 0)
   {
      temp = a;
      a = b;
      b = temp;
   }
}
```

a = 3
b = 6
c = 2
d = 8

(2.3) Read code and answer questions.

```
string foo(int num)
{
  string result = "";
  do {
    result = to_string(num % 8) + result;
      //to_string convert an int to the corresponding string
    num /= 8;
  } while (num != 0);

  return result;
}
```

What are the return for foo(8) and foo(17)?

result = "", num = 8

result = "0", num = 1

result = "10", num = 0

foo(8) = "10";

foo(17) = "21";

result = "", num = 17

result = "1", num = 2

result = "21", num = 0

3. (20 points) **Define a function**, for a given string str, return a string whose letters are the odd-index letters in str with the same order. That is, suppose str is "abc", then return "b".

```
string foo(string str)
{
    string r = "";
    for(int i=1; i<str.length(); i+=2)
        r += str[i];
    return r;
}
```

4. (20 points) Write code inside main function, no need to include libraries.
   (1) Enter numbers a and b, which can contain decimals.
   (2) Solve x for $ax + b^2 = 0$. You need to consider the case when a is zero or not.

```
int main()
{
    double a, b;
    cout << "please enter a";
    cin >> a;
    while (a == 0)
    {
        cout << "please re-enter a";
        cin >> a;
    }
    cout << "please enter b";
    cin >> b;
    double x = -b*b/a;
    cout << x << endl;
    return 0;
}
```