

# CST8110: Introduction to Computing: Midterm Part 1: Fall 2007

Name:

## Multiple Choice Section (12 marks)

Identify the best answer for the following questions. Answer on the mark sense cards. Use only pencil. (Pen markings are not read correctly and the automated marking machine will mark each question as incorrect.)

1. Given the following, what will be in **nFine** after these statements execute:

```
int nFine = 0, nSpeed = 95;
if (nSpeed > 120)
    nFine = 150;
if (nSpeed < 105)
    nFine = 100;
if (nSpeed < 95)
    nFine = 80;
if (nSpeed < 75)
    nFine = 50;
```

- a) 150  
b) 100  
c) 80  
d) 50  
e) 0
2. When you declare a variable, your code:
- a) Announces the name of the variable and its data type (that is, what kind of variable it will be).  
b) Allocates space in memory (probably RAM). The memory will be used to store values.  
c) Gives the variable its starting value.  
d) Loads the variable into the Arithmetic Logic Unit (ALU).  
e) Items a), b) and c).
3. At the level of hardware, when the computer selects a variable to perform some arithmetic operation, its address is uniquely identified by the:
- a) arithmetic logic unit (ALU).  
b) RAM.  
c) control unit.  
d) data bus.  
e) None of the above.
4. You use **#include** to:
- a) Tell the compiler where program execution begins.  
b) Declare and define your variables.  
c) Open the file that's named by the **#include**, and insert the code in your program.  
d) Add comments to your program.  
e) Tell the linker how to assemble all components of your project.
5. Which of the following is NOT a valid **char**?
- a) 'A'  
b) ' ' (a blank space)  
c) '7'  
d) '\n'  
e) None. All are valid **char** values.

6. Given the following, answer the next 3 questions:

```
struct Date {
    int nDay;
    int nMonth;
    int nYear;
};

struct Food {
    int nProductNumber;
    Date dMadeOn;
    Date dBestBefore;
};

Food Yogurt, Cheese;
```

7. Create an object called **CurDate** of type **Date** and initialize it with today's date (which is March 12, 2007).
- a) **CurDate Date.**{ 12, 3, 2007 };  
b) **Date CurDate = { 12, 3, 2007 };**  
c) **Date CurDate = (12, 3, 2007);**  
d) **Date.CurDate = 12, 3, 2007;**  
e) None of above.
8. A batch of cheese was made on April 13<sup>th</sup>, 2006. Set the month member of the manufacturing date for **Cheese** to April.
- a) **Cheese.Date.dMadeOn.nMonth = { 4 };**  
b) **Cheese = { 13, 4, 2006 };**  
c) **Cheese.dMadeOn.nMonth = April;**  
d) **Cheese.dMadeOn.nMonth = 4;**  
e) None of above.
9. To put **2008** into the best before year for **Yogurt**:
- a) **Yogurt.Date.nYear = 2008;**  
b) **Yogurt->dBestBefore->nYear = 2008;**  
c) **Yogurt.nYear.dBestBefore = 2008;**  
d) **Yogurt.dBestBefore.nYear = { 2008 };**  
e) None of above.
10. Which answer identifies the correct size of each data type (in bytes):
- a) **int:2 float:4 long:4 char:1 double:8**  
b) **int:4 float:6 long:4 char:1 double:8**  
c) **int:4 float:6 long:8 char:1 double:6**  
d) **int:2 float:4 long:4 char:2 double:12**  
e) **int:4 float:4 long:4 char:1 double:8**

11. A **for** loop contains a statement that displays the contents of the variable **n**. Identify the answer that results in a display sequence of **1** to **10** (inclusive).

- a) `for (int n = 0; n <= 10; n++)`
- b) `for (int n = 0; n++; n != 10)`
- c) `for (int n = 1; n < 10; n++)`
- d) `for (int n = 1; n <= 11; n++)`
- e) None of above.

12. Which of the following expressions will yield a random number between **-10** and **10** (inclusive)?

- a) `rand( -10 , 10 )`
- b) `rand( ) % 21 - 10`
- c) `rand( ) % 20 - 10`
- d) `rand( ) % -10 + 20`
- e) `rand( -10 → 10)`

13. The expression **40%6** evaluates to:

- a) 2
- b) 34
- c) 6
- d) 4
- e) None of the above.

### Debugging Section (6 marks)

14. (3 marks) Identify the errors in the following code. Note your corrections directly in the source code. There are three categories of error. If a category of error occurs more than once, you must mark each occurrence.

```
using namespace std;

////////////////////////////////////
struct {
    int nModelNumber;
    int nPartNumber;
    float fCost;
} Part;
////////////////////////////////////

void main() ;
{
Part part1;

part1.nModelNumber = 6244;
part1.nPartNumber = 373;
part1.fCost = 217.55F;

cout << "Model " << part1.nModelNumber;
cout << ", Part " << part1.nPartNumber;
cout << ", costs $" << part1.fCost << endl;
}
```

15. (3 marks) Identify the three categories of error in the following code: 2 syntax, 1 logic. Note your corrections directly in the source code on this paper.

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

#include <conio.h>

void main()
{
char nDirection = "a";
int nX = 10, nY = 10;

while (nDirection != "\r") {
    cout << "\n\nYour location is " << nX << ", " << nY;

    if (nX<5 OR nX>15)
        cout << "\nBeware: dragons lurk here";

    cout << "\nEnter direction (n, s, e, w): ";
    nDirection = getche();
    switch(nDirection) {
        case "n": nY--;
        case "s": nY++;
        case "e": nX++;
        case "w": nX--;
    }
}
}
```

### Tracing Program Execution (3 marks)

16. (3 marks) What would display on the screen exactly from this program?

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

void main()
{
int a = 5;
while (a >= 0)
{
if ( (a % 2) != 0 )
cout << a-- << endl;
cout << --a << endl;
}

cout << "****" << endl;
}
```

**Answer**

### Working with structs (8 marks)

17. (4 marks) Using one or more **struct** specifications, organize data about credit cards held by people. Each credit card will store the following: *account number* (8-digits), *month of card expiry* (as a 2-digit value), *year of expiry* (as a 4 digit value), and *balance* (as a floating-point value). Each person will have a string to hold his or her name, and four different credit cards.

**Answer**

19. (2 marks) Write the code to calculate the total balance owing by summing the balances for each of the four credit cards in the person object called **Henry** (the person object created in the previous question).

**Answer**

18. (2 marks) Create a person object called **Henry** (this person object will be built using the **struct** you specified above). Initialize **Henry** with reasonable data values.

**Answer**