

Université d'Ottawa
Faculté de genie

École d'ingénierie et de
technologies de l'information



University of Ottawa
Faculty of Engineering
School of Information
Technology and Engineering

GNG 1106 Fall 2015 Final Exam (English)

- Closed-book, calculators allowed.
- 180 minutes
- Write all your answers in the **exam booklet(s)**, not on the exam paper! Everything you write on the exam paper will be ignored for grading.

Part I: Multiple-Choice Questions (32 Points)

Select ONE answer for each question in this part. Do not justify your answer.

Question 1 (4 points)

```
#include <stdio.h>

void foo(char *s)
{
    *(s+3)='\0';
}

int main()
{
    char myString[100]="hello world";
    foo(myString);
    printf("%s\n", myString);
    return 0;
}
```

Regarding the above program, which one of the following statements is correct?

- A The program does not compile.*
- B The program makes an illegal memory access.*
- C The program prints hello.*
- D The program prints hell.*
- E The program prints hel.*
- F The program prints he.*
- G None of above is correct.*

Solution: E

Question 2 (4 points)

```
#include <stdio.h>

int main()
{
    int a[5]={1, 2, 3, 4, 5};
    int *p;
    p=a+2;
    printf("%d\n", p[1]);
    return 0;
}
```

Regarding the above program, which one of the following statements is correct?

- A The program does not compile.*
- B The program makes an illegal memory access.*

- C The program prints 1.
- D The program prints 2.
- E The program prints 3.
- F The program prints 4.
- G The program prints 5.
- H None of above is correct.

Solution: F

Question 3 (4 points)

```
#include <stdio.h>

int main()
{
    int a[5]={1, 2, 3, 4, 5};
    int b[5];
    b=a;
    printf("%d\n", b[1]);
    return 0;
}
```

Regarding the above program, which one of the following statements is correct?

- A The program does not compile.
- B The program makes an illegal memory access.
- C The program prints 1.
- D The program prints 2.
- E The program prints 3.
- F The program prints 4.
- G The program prints 5.
- H None of above is correct.

Solution: A

Question 4 (4 points)

```
#include <stdio.h>

struct widget
{
    int n;
    int *p;
};
```

```

int main()
{
    int a[5]={1, 2, 3, 4, 5};
    struct widget w;
    w.n=5;
    w.p=a;
    w.p[3]=11;
    printf("%d\n", a[3]);
    return 0;
}

```

Regarding the above program, which one of the following statements is correct?

- A The program does not compile.*
- B The program makes an illegal memory access.*
- C The program prints 3.*
- D The program prints 4.*
- E The program prints 11.*
- F None of above is correct.*

Solution: E

Question 5 (4 points)

```

#include <stdio.h>

struct widget
{
    int n;
    int *p;
};

struct widget getWidget()
{
    struct widget u;
    int a[5]={1, 2, 3, 4, 5};
    u.n=5;
    u.p=a;
    return u;
};

void printWidget(struct widget w)
{
    int i;
    for (i=0; i<w.n; i++)
        printf("%d:", w.p[i]);
    printf("\n");
}

```

```

int main()
{
    int i;
    struct widget w;
    w=getWidget();
    for (i=0; i<5; i++)
        w.p[i]=11;
    printWidget(w);
    return 0;
}

```

Regarding the above program, which one of the following statements is correct?

- A The program does not compile.*
- B The program makes an illegal memory access.*
- C The program prints 1:2:3:4:5:.*
- D The program prints 11:11:11:11:11:.*
- E None of above is correct.*

Solution: B

Question 6 (4 Points)

```

#include <stdio.h>

void increaseBy2(int x)
{
    x=x+2;
}

int main()
{
    int a=10;
    increaseBy2(a);
    printf("%d\n", a);
    return 0;
}

```

Regarding the above program, which one of the following statements is correct?

- A The program does not compile.*
- B The program makes an illegal memory access.*
- C The program prints 10.*
- D The program prints 12.*
- E None of above is correct.*

Solution: C

Question 7 (4 Points) *What will be the most likely print-out of the following statement?*

```
printf("%d\n", (rand()==rand()));
```

A 1.

B 0.

C a random number

D None of above.

Solution: B

Question 8 (4 Points)

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

```
int main()
{
    int tab[5]={4,2,0,1,3};
    int i, tab2[5];
    for (i=0; i<5; ++i)
        tab2[tab[i]]=tab[i];
    for (i=0;i<5;++i)
        printf("%d", tab2[i]);

    return 0;
}
```

The above program prints

A 24301

B 01432

C 01234

D 43201

E 42013

F None of above

Solution: C

Part II: Code Tracing Questions (12 Points)

Give the print-out of each program in this part. You are not required to justify your answers. But if you do, your justifications will be considered for grading.

Question 9 (6 Points)

```
#include <stdio.h>

int main()
{
    int i=0, j=0, count=0;
    for (i=0; i<4; i++)
    {
        for (j=i+count/2; j<10; j++)
        {
            printf("*");
            count++;
        }
        printf("\n");
    }
    printf("done\n");
    return 0;
}
```

Solution:

```
*****
*****
*
```

done

Grading criteria: Grade this question by adding marks in this order:

- The print-out contains 5 lines : +2 points
- The first line is correct, i.e., being 10 stars: +2 points
- Given the first line is correct, the second line is correct: +2 points
- Given the first two lines are correct, the third line is correct: +2 points
- Given the first three lines are correct, the fourth line is correct: +2 points
- Given the first four lines are correct, the fifth line is correct: +2 points

Question 10 (6 Points)

```
#include <stdio.h>

int getNext()
{
    static int counter=0;
    static int sign=1;
```

```

    static int prev=3;
    static int cur=4;
    int out;

    counter++;
    if (counter %2==0)
        sign=sign*(-1);
    out=cur*sign+prev;
    prev=cur;
    cur=out;
    return out;
}

int main()
{
    int i;
    for (i=0;i<7;i++)
        printf("%d\n", getNext());
    return 0;
}

```

Solution:

```

7
-3
10
7
17
-10
27

```

Grading criteria:

- This question is graded also by adding marks: except for the first line and last line each worth +1 point, every other line is worth +2 points. Similar to the above question, a student only gets the mark for a line if his answers for all lines prior to it is correct.
- Consider giving the student up to 3 points if the student gets 3 or fewer points according to the above criteria but he is able to demonstrate (in the justification of his answer) understanding static variables.

Part III: Programming Questions (56 Points)

It is sufficient to give the required code for each question in this part. But if you choose to give pseudo-code or flow-chart to show your design and ideas, it may help you getting partial marks in case the code is incomplete or incorrect.

First grading criteria:

If from the student's code, it is obvious that the student has never independently written a program, directly give 0 to the question. Apply this rule to all question, and apply this rule first.

Question 11 (13 Points) *Write a program that computes the average of the set of non-negative numbers that the user enters. More precisely, the program will keep prompting the user to enter a number until the user enters a negative number. If the first number the user enters is a negative number, the program prints no non-negative number is entered and exits; otherwise the program prints the average of the non-negative numbers that the user has entered.*

Grading Criteria: Apply the following criteria in order:

- The student having perfectly correct logic (demonstrated via flow-chart or pseudo-code) gets at least 6 points.
- Grade the question by deducting points:
 - syntax error: -1 point each; the same syntax error appearing twice only gets points deducted once; obvious non-essential typos are not considered as syntax error.
 - logical error: -2 points each; maximal deductible mark for logical errors is 6 points.
- For students getting 3 or fewer points using the above criteria, consider the “adding points” criterion to see if he can get better marks:
 - Using this criterion, the maximal mark the student can get is 4 points.
 - Knowing how to use scanf, printf, loops, compute the sum and average, properly set up conditions for stopping prompting user: each +1 point

Question 12 (15 points) *Write a function that takes an array of int typed values as input such that when the function is called, the calling function gets the largest value, smallest value and the sum of the elements in the array. You are free to design the prototype of the function (i.e., free to design the input and output variables of the function). Also write a main function which calls this function and demonstrate the use of it.*

Grading Criteria: Apply the following criteria in order:

- The code for the main function is worth 2 points. The code for the function is worth 13 points.
- Grading the code for the function (13 points):
 - The student who has the correct prototype, namely, knowing how to send the required information back to the calling function gets at least 4 points.
 - The student who has the correct prototype and the correct logic gets at least 6 points.
 - Deducting points criteria:

- * syntax error: -1 point each; the same syntax error appearing twice only gets points deducted once; obvious non-essential typos are not considered as syntax error.
 - * logical error (including prototype error): -2 points each; maximal deductible mark for logical errors is 8 points.
- For student getting 3 points or lower using the above criteria, consider the “adding points” criterion to see if he can get better marks:
 - Using this criterion, the maximal mark the student can get is 4 points.
 - Knowing how to use array, finding largest or smallest, correct use of pointers: each +2 point
 - correct use of scanf, printf and loops are NOT given points (its been given marks in the grading of previous question)

Question 13 (15 Points) *A structure studentRecord is defined as below, which is used to represent the record of a student in a course.*

```
struct studentRecord
{
    char lastName[100];
    char firstName[100];
    int studentId;
    double mark;
};
```

Implement a function with the following prototype:

```
void writeRecordsToFile(struct studentRecord *records, int numOfRecords, char *filename);
```

The function writes an array of studentRecord-typed data to a specified file. More specifically, the input variable records specifies the address pointing to the input array; the input variable numOfRecords specifies the number of student records in the array; and filename specifies (the address of) the file name string of the file which will be created and written into.

In addition, the function calculates the letter grade (i.e. A+, A, B, etc) from the mark of the student and writes the letter grade to the file. We assume the letter grade is calculated as follows:

*A+ if mark between 90 and 100
 A if mark between 80 and 89
 B if mark between 70 and 79
 C if mark between 60 and 69
 D if mark between 50 and 59
 F if mark is below 50*

Also write a main function which calls this function and demonstrate the use of it. The main function will prompt the user to enter the student?s information until all the students have been entered.

Here is a sample of the output file:

Lastname	Firstname	StudentID	Mark	LetterGrade
Smith	John	111	83	A
Harp	Joe	222	92.8	A+
Lepage	Pete	333	75.6	B

Grading Criteria

Grade this question by adding points:

- Understanding structure and using correct notations/commands for structures: +5 points
- Knowing how to write to files: +4 points
- Having the correct logic and code for converting numerical grades to letter grades: +2 points
- Everything else: +4 points.

Question 14 (13 Points) *Write a function that takes an array of int-typed values as input such that when the function is called, it prints all distinct numbers in the array in ascending order. For example, if the input array to the function is {3, 4, 2, 3, 9, 6, 4, 2, 5, 2}, the function will print*

2 3 4 5 6 9

You are free to design the prototype of the function. Also feel free to design additional functions that are called by this function, to make your code better modularized.

Grading Criteria

Grade this question by adding points:

- Logic: + 9 points, see below for break-down.
 - knowing how to sort: +3 points:
 - knowing how to get the unique numbers in the array: +4 points
 - the rest: +2 points
- Implementation: + 4 points