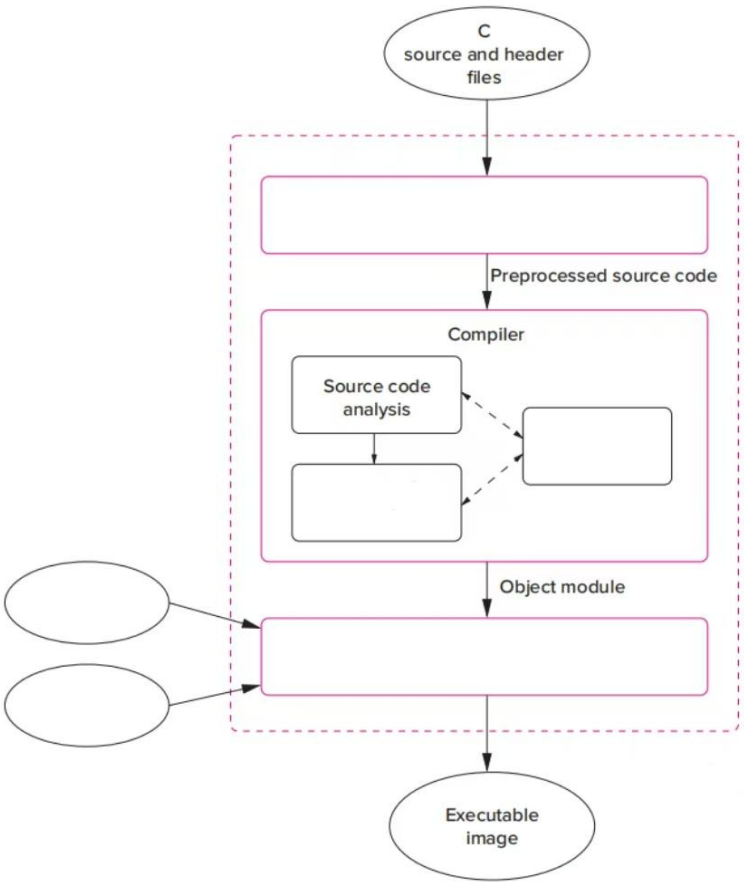
Homework 7

T1

Both C and C++ are compiled languages. The C or C++ compiler follows the typical mode of translation from a source program to an *executable image*. An executable image is a machine language representation of a program that is ready to be loaded into memory and executed. The compilation mechanism involves several distinct com ponents, notably the preprocessor, the compiler itself, and the linker.

Please complete the following image and illustrate the overall compilation process of C.



T2

Another advantage of compilation over interpretation is that a compiler can optimize code more thoroughly. Since a compiler can examine the entire program when generating machine code, it can reduce the amount of computation by analyzing what the program is trying to do.

The following algorithm performs some very straightforward arithmetic based on values typed at the keyboard. It outputs a single result.

1. Get W from the keyboard
2. X<-W+W

3. Y<-X+X
4. Z<-Y+Y
5. Print Z to the screen

a. An interpreter would execute the program statement by statement. In total, five statements would execute. If the underlying ISA were capable of all arithmetic operations (i.e., addition, subtraction, multiplication, division), at least how many operations would be needed to carry out this program? State what the operations would be.

b. A compiler would analyze the entire program before generating machine code, and it would possibly optimize the code. If the underlying ISA were capable of all arithmetic operations (i.e., addition, subtraction, multiplication, division), at least how many operations would be needed to carry out this program? State what the operations would be.

T3

Given that a and b are both integers equal to the values 6 and 9, respectively, what is the value of each of the following expressions?
Also, if the value of a or b changes, provide their new value.

a. a|b
b. a || b
c. a&b
d. a && b
e. !(a + b)
f. a%b
g. b/a
h. a=b
i. a=b=5
j. ++a + b--
k. a = (++b < 3) ? a : b
l. a <<= b

Expression	**Value of expression**	**Value of a afterwards**	**Value of b afterwards**
a | b			
a || b			
a & b			
a && b			
!(a + b)			
a % b			
b / a			
a = b			
a = b = 5			
++a + b—			
a = (++b < 3) ? a : b			
a <<= b			

T4

Explain the differences between the following C statements:

a. j = i++;

b. j = ++i;

c. j = i + 1;

d. i += 1;

e. j = i += 1;

f. Which statements modify the value of i? Which ones modify the value of j? If i=0 and j=1 initially, what will the values of i and j be after each statement is run separately?

T5

At least how many times will the statement called loopBody execute the following constructs?

a. while (condition)
loopBody;

b. do
loopBody;
while (condition);

c. for (init; condition; update)
loopBody;

d. while (condition1)
for (init; condition2; reinit)
loopBody;

e. do
do
loopBody;
while (condition1);
while (condition2);

T6

Provide the output of each of the following code segments.

a.
```
int x = 20;
int y = 10;
while ((x > 10) && (y & 15)) {
y = y + 1;
x = x - 1;
printf("*");
}
```

b.
```
for (int x = 10; x ; x = x - 1)
printf("*");
```

c.
```
for (int x = 0; x < 10; x = x + 1)
```

```
if (x % 2)
printf("*");
```

d.
```
int x = 0;
while (x < 10) {
for (int i = 0; i < x; i = x + 1)
printf("*");
x = x + 1;
}
```

T7

The following C program uses a combination of global variables and local variables with different scope. What is the output?

```
#include <stdio.h>
int t = 1; // Global variable
int sub1(int fluff);
int main (void)
{
    int t = 2;
    int z;
    z = t;
    z = z + 1;
    printf("A: The variable z equals %d\n", z);
    {
        z = t;
        t = 3;
        {
            int t = 4;
            z = t;
            z = z + 1;
            printf("B: The variable z equals %d\n", z);
        }
        z = sub1(z);
        z = z + 1;
        printf("C: The variable z equals %d\n", z);
    }
    z = t;
    z = z + 1;
    printf("D: The variable z equals %d\n", z);
}
int sub1(int fluff)
{
```

```
    int i;
    i = t;
    return (fluff + i);
}
```

T8

The following code reads a string from the keyboard and prints out a version with any uppercase characters converted to lowercase.
However, it has a flaw. Identify it.

```
#include <stdio.h>
#define MAX_LEN 10
char *LowerCase(char *s);

int main(void)
{
    char str[MAX_LEN];
    printf("Enter a string : ");
    scanf("%s", str);
    printf("Lowercase: %s \n", LowerCase(str));
}
char *LowerCase(char *s)
{
    char newStr[MAX_LEN];
    for (int index = 0; index < MAX_LEN; index++) {
        if ('A' <= s[index] && s[index] <= 'Z')
            newStr[index] = s[index] + ('a' - 'A');
        else
            newStr[index] = s[index];
    }
    return newStr;
}
```

T9

Consider the following program:

```
#include <stdio.h>
int main(void)
{
int x = 0;
int y = 0;
char label[10];
```

```
scanf("%d %d", &x, &y);
scanf("%s", label);
printf("%d %d %s\n", x, y, label);
}
```

a. What gets printed out if the input stream is 46 29 BlueMoon?

b. What gets printed out if the input stream is 46 BlueMoon?

c. What gets printed out if the input stream is 111 999 888?

T10

The following C program is compiled into the LC-3 machine language and executed. The run-time stack begins at xEFFF. The user types the input abac followed by a return.

```
#include <stdio.h>
#define MAX 4
typedef Rec
struct rec_t {
        char ch;
        struct Rec *back;
};
int main(void)
{
        struct Rec *ptr, pat[MAX+2];
        int i = 1, j = 1;
        printf("Pattern: ");
        pat[1].back = pat;
        ptr = pat;
        while ((pat[i].ch = getchar()) != '\n') {
                ptr[++i].back = ++ptr;
                if (i > MAX) break;
        }
        while (j <= i)
                printf("%d ", pat[j++].back - pat);
                // Note the pointer arithmetic here: subtraction
                // of pointers to structures gives the number of
                // structure elements, not the number
                // of memory locations
}
```

a. Show the contents of the stack frame for main when the program terminates.

b. What is the output of this program for the input abac?