

WEEK-END ASSIGNMENT-08

C Storage Classes & Recursion

Operating Systems Workshop (CSE 3541)

Problem Statement:

Working with different storage classes and Experiment with one of the powerful tool, recursion, in problem solving and programming.

Assignment Objectives:

To learn about storage classes and get the idea of how function calls itself to solve computational problem.

Instruction to Students (If any):

Students are required to write his/her own program by avoiding any kind of copy from any sources. Additionally, They must be able to realise the outcome of that question in relevant to systems programming. You may use additional pages on requirement.

Programming/ Output Based Questions:

1. Consider the following ANSI C program;

```
#include <stdio.h>
int main()
{
    static int i=5;
    if(--i){
        main();
        printf("%d ",i);
    }
    return 0;
}
```

What is the output of the above program?

Output with explanation

2. Consider the following ANSI C program;

```
#include <stdio.h>
int a, b, c = 0;
void prtFun(void);
int main()
{ static int a = 1; /* Line 1 */
  prtFun( );
  a+=1;
  prtFun( );
  printf("\n %d %d ", a, b);
  return(0);
}
void prtFun(void)
{ static int a = 2; /* Line 2 */
  int b = 1;
  a + = ++b;
  printf(" \n %d %d ", a, b);
}
```

What is the output of the above program?

Output with explanation

3. Consider the following ANSI C program;

```
#include <stdio.h>
int a, b, c = 0;
void prtFun(void);
int main()
{ auto int a = 1; /* Line 1 */
  prtFun( );
  a+=1;
  prtFun( );
  printf("\n %d %d ", a, b);
  return(0);
}
void prtFun(void)
{ register int a = 2; /* Line 2 */
  int b = 1;
  a + = ++b;
  printf(" \n %d %d ", a, b);
}
```

What is the output of the above program?

Output with explanation

4. What is printed by the following ANSI C program?

[GATE 2005]

```
#include<stdio.h>
int f(int n, int k){
    if(n==0) return 0;
    else if(n%2) return f(n/2, 2*k)+k;
    else return f(n/2, 2*k)-k;
}
int main(){
    printf( "%d",f(20,1));
    return 0;
}
```

Output with explanation

5. What is printed by the following ANSI C program?

[GATE 2007]

```
#include<stdio.h>
void f(int n){
    if(n<=1){
        printf("%d",n);
    }
    else{
        f(n/2);
        printf("%d",n%2);
    }
}
int main()
{
    f(173);
    return 0;
}
```

Output with explanation▼

6. What is printed by the following ANSI C program?

[GATE 2011]

```
#include<stdio.h>
unsigned int foo(unsigned int n, unsigned int r)
{
    if(n>0) return((n%r)+foo(n/r,r));
    else return 0;
}
int main()
{
    printf("%d\n", foo(345,10));
    return 0;
}
```

Output▼

7. What is printed by the following ANSI C program?

```
#include<stdio.h>
void print(int n)
{
    printf("Hello ");
    if(n++ == 0) return ;
    print(n);
    n++;
}

int main()
{
    print(-4);
}
```

Output▼

8. Consider the program below;

```
#include<stdio.h>
int foo(int a){
    if( a == 0 || a ==2) return 1;
    return (foo(--a) * (a--));
}
int main()
{
    printf("%d\n", foo(4));
}
```

Output▼

9. Consider the following C program

```
#include<stdio.h>
int main(){
    register int a =10;
    int *ptr = NULL;
    ptr = &a;
    *ptr = 5;
    printf("%d", *ptr);
    return(0);
}
```

Find the error in the program with proper reasoning

Output▼

10. Consider the following C function;

file1.c

```
-----  
extern int count;  
void write_extern() {  
    count +=2;  
}
```

file2.c

```
-----  
#include<stdio.h>  
#include "file1.c"  
int count = 5;  
int main() {  
    write_extern();  
    write_extern();  
    printf("%d\n", count);  
    return(0);  
}
```

Find the output if “file2.c” is compiled and executed:

Output with explanation▼

11. Write the output of the following program;

```
#include<stdio.h>  
int i=5;  
int main()  
{  
    extern int j;  
    printf("\ni=%d \nj=%d", i, j);  
    int j=10;  
    return 0;  
}  
  
int j =10;
```

Output▼

12. Find the output and different types of pointer involved in the code snippet;

```
#include <stdio.h>  
int fun(int num){  
    while(num>0)  
        num=num*fun(num-1);  
    return num;  
}  
int main(){  
    int x=fun(8);  
    printf("%d",x);  
    return 0;  
}
```

Output▼

13. Write a program to find the sum of an array elements using recursion.

Program and Output▼

```
#include <stdio.h>

int arraySum(int arr[], int n) {
    if (n == 0) {
        return 0;
    } else {
        return arr[n - 1] + arraySum(arr, n - 1);
    }
}

int main() {
    int n;

    printf("Enter the size of the array: ");
    scanf("%d", &n);

    int arr[n];

    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }

    int sum = arraySum(arr, n);
    printf("Sum of array elements: %d\n", sum);

    return 0;
}
```

```
Enter the size of the array: 5
Enter the elements of the array:
1 2 3 4 5
Sum of array elements: 15
```

14. Write a program to print “n” Fibonacci numbers using recursion.[N.B: The program format should be as follows]

```
#include <stdio.h>
... print_fibo(.....){
    ...
    ...
    ...
}
... main(){
    // get data from user
    print_fibo(...); // to print elements
}
```

Program and Output▼

```
#include <stdio.h>

// Function to calculate and print Fibonacci numbers using recursion
void printFibo(int n, int a, int b, int count) {
    if (count > n) {
        return;
    }

    printf("%d ", a);
    printFibo(n, b, a + b, count + 1);
}

int main() {
    int n;

    printf("Enter the number of Fibonacci numbers to print: ");
    scanf("%d", &n);

    printf("Fibonacci series of %d numbers: ", n);
    printFibo(n, 0, 1, 1);

    return 0;
}
```

```
Enter the number of Fibonacci numbers to print: 8
Fibonacci series of 8 numbers: 0 1 1 2 3 5 8 13
```

15. Write a program to print the binary equivalent of a Decimal number using recursion.

Program and Output▼

```
#include <stdio.h>

// Function to convert decimal to binary using recursion
void decimalToBinary(int decimal) {
    if (decimal > 0) {
        decimalToBinary(decimal / 2);
        printf("%d", decimal % 2);
    }
}

int main() {
    int decimal;

    printf("Enter a decimal number: ");
    scanf("%d", &decimal);

    printf("Binary equivalent: ");
    if (decimal == 0) {
        printf("0");
    } else {
        decimalToBinary(decimal);
    }

    printf("\n");

    return 0;
}
```

```
Enter a decimal number: 25
Binary equivalent: 11001
```

16. Write a program to remove adjacent duplicate characters from a string using recursion.

Program and Output▼

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

// Function to remove adjacent duplicates from a string using recursion
void removeAdjacentDuplicates(char str[], int currentIndex) {
    if (str[currentIndex] == '\0') {
        return;
    }

    if (str[currentIndex] == str[currentIndex + 1]) {
        // Shift characters to remove the duplicate
        for (int i = currentIndex; str[i] != '\0'; i++) {
            str[i] = str[i + 1];
        }
        removeAdjacentDuplicates(str, currentIndex);
    } else {
        // Move to the next character
        removeAdjacentDuplicates(str, currentIndex + 1);
    }
}

int main() {
    char inputString[100];

    printf("Enter a string: ");
    fgets(inputString, sizeof(inputString), stdin);

    // Remove the newline character from input
    inputString[strcspn(inputString, "\n")] = '\0';

    printf("Original string: %s\n", inputString);

    // Call the function to remove adjacent duplicates
    removeAdjacentDuplicates(inputString, 0);

    printf("String after removing adjacent duplicates: %s\n", inputString);

    return 0;
}
```

```
Enter a string: programming
Original string: programming
String after removing adjacent duplicates: progaming
```


17. Write a program to find the sum of a geometric sequence using recursion.

Program and Output▼

```
#include <stdio.h>

// Function to find the sum of a geometric sequence using recursion
double geometricSum(double firstTerm, double commonRatio, int n) {
    if (n == 0) {
        return 0;
    } else {
        return firstTerm + geometricSum(firstTerm * commonRatio, commonRatio, n - 1);
    }
}

int main() {
    double firstTerm, commonRatio;
    int n;


    printf("Enter the first term of the geometric sequence: ");
    scanf("%lf", &firstTerm);

    printf("Enter the common ratio of the geometric sequence: ");
    scanf("%lf", &commonRatio);

    printf("Enter the number of terms in the sequence: ");
    scanf("%d", &n);

    // Calculate and print the sum using recursion
    double sum = geometricSum(firstTerm, commonRatio, n);
    printf("Sum of the geometric sequence: %.2lf\n", sum);

    return 0;
}
```



```
Enter the first term of the geometric sequence: 2
Enter the common ratio of the geometric sequence: 3
Enter the number of terms in the sequence: 4
Sum of the geometric sequence: 62.00
```

18. Write a recursive function that takes n words as input and print them in reverse order on separate lines.
The prototype of the function should be as follows:
void reverse_input_words(int n)

Program and Output▼

```
#include <stdio.h>

// Function to reverse input words recursively
void reverseInputWords(int n) {
    if (n == 0) {
        return;
    } else {
        char word[50];
        // Get input word
        printf("Enter word %d: ", n);
        scanf("%s", word);

        // Recursively call the function for the remaining words
        reverseInputWords(n - 1);

        // Print the word in reverse order
        printf("%s\n", word);
    }
}

int main() {
    int n;

    printf("Enter the number of words: ");
    scanf("%d", &n);

    // Call the function to reverse input words
    reverseInputWords(n);

    return 0;
}
```

```
Enter the number of words: 3
Enter word 1: Apple
Enter word 2: Banana
Enter word 3: Orange
Orange
Banana
Apple
```